

UNIVERSIDADE FUMEC
FACULDADE DE CIÊNCIAS EMPRESARIAIS – FACE

LUIS FELIPE CORRÊA PÉREZ

**AVALIAÇÃO DE PRODUTIVIDADE PHP X JAVA NO DESENVOLVIMENTO DE
SISTEMAS DE SOFTWARE**

BELO HORIZONTE
2010

UNIVERSIDADE FUMEC

LUIS FELIPE CORRÊA PÉREZ

**AVALIAÇÃO DE PRODUTIVIDADE PHP X JAVA NO DESENVOLVIMENTO DE
SISTEMAS DE SOFTWARE**

Monografia realizada na Universidade FUMEC, no curso de Ciência da Computação, apresentada à disciplina Trabalho de Conclusão de Curso.

Orientadores:

Professor Osvaldo Manoel Corrêa

Professor Ricardo Terra Nunes Bueno
Villela

BELO HORIZONTE

2010

RESUMO

Com a explosão em popularidade da *World Wide Web* em 1993, diversas linguagens de programação voltadas a esse novo ambiente foram criadas. Dentre tais linguagens, PHP e Java merecem destaque especial devido a sua popularidade. No entanto, existe uma questão recorrente enfrentada por empresas desenvolvedoras de sistemas de software: Qual linguagem de programação proporciona uma melhor relação custo/benefício? Diante disso, este estudo provê um estudo comparativo entre PHP e Java. Inicialmente, são apresentados conceitos, princípios, características dessas linguagens, incluindo detalhes de seu ambiente e perfil de seus desenvolvedores. Em seguida, é avaliada a produtividade do desenvolvimento – em PHP e em Java EE – de um sistema Web hipotético denominado XMovies. Essa avaliação tem por objetivo demonstrar vantagens e desvantagens de cada linguagem e prover uma análise comparativa a fim de apontar qual das linguagens de programação é mais adequada no desenvolvimento de sistemas Web convencionais.

Palavras-chave: Avaliação de Produtividade. Desenvolvimento de sistemas. Java EE. PHP.

ABSTRACT

The gain popularity of the World Wide Web in 1993 has motivated the creation of several programming languages directed to this new environment. Among such languages, PHP and Java deserve special attention due to their popularity. However, there is a recurrent issue faced by software development companies: What is the most cost-effective programming language? Thus, this study provides a comparative study between PHP and Java. Initially, we present the concepts, principles, and characteristics of both languages, including details of their environment and their developer profile. Next, we evaluate the software development productivity – in PHP and Java EE – of a hypothetical Web system called XMovie. This evaluation aims to demonstrate the advantages and disadvantages of each language and to provide a comparative analysis in order to point out which programming language fits better to the development of conventional Web systems.

Keywords: Evaluation of Productivity. Systems development. Java EE. PHP.

SUMÁRIO

INTRODUÇÃO	5
1. PHP	7
1.1 CARACTERÍSTICAS.....	8
1.2 AMBIENTE DE DESENVOLVIMENTO	10
1.3 PERFIL DO DESENVOLVEDOR	12
1.4 APLICAÇÕES.....	13
2. JAVA EE.....	14
2.1 CARACTERÍSTICAS.....	15
2.2 AMBIENTE DE DESENVOLVIMENTO	16
2.3 PERFIL DO DESENVOLVEDOR	17
2.4 APLICAÇÕES.....	18
3. ESTUDO DE CASO / AVALIAÇÃO DE PRODUTIVIDADE.....	19
3.1 SISTEMA MOTIVADOR.....	19
3.2 IMPLEMENTAÇÃO	20
3.2.1 AVALIAÇÃO PHP	22
3.2.2 AVALIAÇÃO JAVA EE	23
3.3 ANÁLISE	25
3.4 AMEAÇAS À VALIDADE DA AVALIAÇÃO.....	26
4. CONCLUSÃO	27
REFERÊNCIAS.....	30
5. APÊNDICE A – Especificação de Requisitos do projeto XMovies	32
6. APÊNDICE B – Código fonte PHP	60
7. APÊNDICE C – Código fonte Java EE	95

INTRODUÇÃO

O trabalho de um desenvolvedor de sistemas de software começou no início do século XIX com Ada Lovelace escrevendo o primeiro programa de computador do mundo.

Desde então, o trabalho do desenvolvedor de sistemas veio tornando-se mais complexo e árduo. Novas linguagens de programação surgiram e, com elas, novos sistemas de software. Os novos caminhos abertos obrigam a necessidade de uma especialização e foco do desenvolvedor. Em 1993, quando o mundo conheceu a *World Wide Web*, surgiu um novo foco para o desenvolvimento de sistemas de software. A Web passou a ser uma “plataforma” que demanda cada vez mais pesquisas com foco na sua evolução.

Com isso, surgiram linguagens de programação focadas em Web assim como Java EE, PHP, ASP, .Net etc. E agora? Os desenvolvedores necessitam, mais uma vez, de uma especialização e uma linguagem foco. Todas essas linguagens ainda são muito utilizadas e ainda existem outras. Como decidir? O objetivo deste estudo é ajudar e orientar esses desenvolvedores na escolha da linguagem ideal para o desenvolvimento de sistemas.

Escolhemos duas entre as linguagens mais utilizadas – PHP e Java EE – para analisarmos o desempenho de produtividade de um desenvolvedor de sistemas e prover um estudo de qual linguagem pode ajudar mais no avanço da Web.

A escolha das linguagens Java e PHP se deu devido aos seguintes motivos: (i) são linguagens gratuitas e largamente utilizada no desenvolvimento de sistemas web; (ii) atualmente, ambas são orientadas a objetos e possuem grande semelhança com a linguagem C; (iii) existem a praticamente o mesmo tempo; etc. Convém mencionar também que o autor conhece e trabalha profissionalmente com essas linguagens há mais de 5 anos.

O PHP foi criado por Ramus Lerdorf em 1994 com o objetivo de monitorar o acesso a seu currículo na Internet. Dois estudantes, Andi e Zeev, que utilizavam a linguagem em um projeto acadêmico começaram a colaborar com a linguagem e além de re-escreverem o código-fonte para otimizá-lo, re-escreveram todo o núcleo da linguagem para lançar a quarta versão da linguagem. Atualmente, o PHP encontra-se na quinta versão.

Já o Java EE surgiu de um projeto de pesquisa corporativa interna financiada pela *Sun Microsystems* adquirida pela *Oracle Corporation*. O projeto seria cancelado se a *World Wide Web* não caísse em popularidade em 1993. Atualmente, o Java EE é uma linguagem muito estável e confiável sendo utilizada em vários projetos. A linguagem encontra-se na versão 6.

Assim, este trabalho de conclusão de curso está organizado conforme a seguir. No Capítulo 1, é apresentado o conceito da linguagem de programação *Hypertext Preprocessor* (PHP), seus princípios de funcionamento, seu ambiente de utilização e o perfil do desenvolvedor dessa linguagem. Da mesma forma, o Capítulo 2 apresenta conceitualmente, princípios de funcionamento, ambiente de utilização e características do desenvolvedor da linguagem Java EE. No Capítulo 3, baseado em um sistema motivador, é apresentada uma avaliação e a análise da produtividade de desenvolvimento nas linguagens PHP e Java em sistemas convencionais.

1. PHP

O *Hypertext Preprocessor* (PHP), lançado inicialmente como *Personal Home Pages/Forms Interpreter* (PHP/FI), é uma linguagem de programação criada em 1994, por Ramus Lerdorf. Essa linguagem teve sua origem na necessidade de Ramus Lerdorf em monitorar o acesso a seu currículo na Internet, segundo Dall'Oglio (2009).

Em 1995, Lerdorf disponibilizou seu código na Internet esperando receber ajuda no desenvolvimento da linguagem e correção dos bugs atuais. O projeto começou a ficar popular na Internet, pelas facilidades e funcionalidades que apresentava e centenas de usuários começaram a utilizar em suas páginas, conforme Milani (2010).

No final do ano de 1997, foi lançada a segunda versão da linguagem junto com o novo nome, *Personal Home Pages*(PHP). Segundo Dall'Oglio(2009), estima-se que 50 mil domínios ou 1% da Internet já utilizavam PHP. No mesmo ano, dois estudantes, Andi Gutmans e Zeev Suraski, que utilizavam a linguagem em um projeto acadêmico, começaram a colaborar com o projeto e aprimoraram o PHP. Para tal, os estudantes reescreveram todo o código fonte dando início ao PHP 3, disponibilizado em junho de 1998.

Dall'Oglio (2009) afirma que, no final de 1998, o PHP atingiu aproximadamente 10% dos domínios da Internet. Nessa época, o significado do nome PHP mudou para *Hypertext Preprocessor*. Logo após o lançamento da terceira versão, Andi e Zeev começaram a trabalhar na reescrita do núcleo do PHP para melhorar o desempenho e a modularidade em aplicações complexas.

Com a conclusão do novo núcleo, nomeado como *Zend Engine* (Zeev + Andi), foi publicada e divulgada a quarta versão do PHP. Milani (2010) acredita que 20% dos domínios na Internet utilizavam PHP. Com a tecnologia relativamente estável, apenas em 2004 foi lançada a nova versão PHP 5, que incluía correções dos *bugs* encontrados entre o período de 1998 a 2004 e algumas novas características como novo modelo de orientação a objetos, adição dos modificadores de acessos (*public*, *private* e *protected*), constantes para classes, iteradores, padronização dos métodos construtores e destrutores etc.

1.1 CARACTERÍSTICAS

O PHP é uma linguagem interpretada, ou seja, não utiliza arquivos compilados. Isso significa que o código fonte é interpretado toda vez que uma requisição é feita ao servidor Web. A desvantagem desse modelo, é que, segundo Milani (2010), permite o acesso ao código fonte pelo administrador do servidor o que pode ser camuflado em outras linguagens (Java, por exemplo).

Atualmente, “a grande maioria dos servidores de hospedagem e uso em domínios de Internet está voltada para o PHP” (MILANI, 2010), comprovando o crescente uso da linguagem desde sua criação. Ainda, Milani (2010) garante que PHP é uma tecnologia de ponta para atender praticamente todas as demandas.

O PHP é facilmente instalado nas plataformas Windows e Linux. No Windows, podemos escolher entre alguns servidores Web, entre eles o *Microsoft Internet Information Service (IIS)* e *ApacheHTTP Server*. No Linux, o PHP e o Apache são nativos e, normalmente, são instalados automaticamente no ato de instalação do Sistema Operacional.

As principais características do PHP, segundo Niederauer (2010) são:

1. Gratuidade e código aberto;
2. Pode ser embutido no código HTML (*HyperText Markup Language*);
3. É interpretado pelo servidor, ou seja, o servidor é responsável pelo trabalho de interpretar o código e retornar apenas o código HTML para o usuário final;
4. Compatibilidade com os Sistemas Gerenciadores de Banco de Dados (SGBD) mais utilizados;
5. E portabilidade, permitindo que a aplicação não se limite a um único sistema operacional.

A gratuidade do PHP é uma característica importante, porque garante o crescimento e divulgação da linguagem e mantém a necessidade de atualização na mesma. Segundo Niederauer (2010), é possível encontrar as versões mais recentes no *site* oficial do PHP e recomenda que sempre obtenha a versão mais atual, para

aproveitar os novos recursos e instalar as correções para os defeitos da versão anterior.

O código PHP pode ser inserido entre os códigos HTML garantindo a facilidade de utilizar a linguagem. Porém, Niederauer (2010), indica que seja melhor desenvolver toda parte visual do sistema separadamente e, quando concluído, efetue a inserção do código PHP. Isso porque algumas ferramentas, como Microsoft FrontPage, podem não reconhecer seu código final - PHP e HTML - por não considerar válida a estrutura e desorganizá-lo.

Como dito anteriormente, o PHP é interpretado pelo servidor. A grande vantagem para o usuário final é que, diferentemente da linguagem JavaScript, ao acessar uma página em PHP não há consumo de recursos do computador pessoal para interpretar a página. Outra vantagem é que o código da página não pode ser visualizado ou acessado pelo usuário final já que o servidor retorna apenas a codificação em HTML, conforme Niederauer (2010).

Nos sistemas atuais, o uso um SGBD é praticamente inevitável, principalmente por suas facilidades. SGBDs como Oracle, MySQL, Microsoft SQL Server, PostgreSQL são alguns suportados pelo PHP 5 e ainda possui suporte a uma variada gama de outros SGBDs. Os SGBDs garantem a organização dos dados para que o PHP possa encontrá-los e manipulá-los.

Dentre as principais vantagens, destaca-se a portabilidade. Com a abertura do código fonte do PHP, vários desenvolvedores criaram suas próprias versões do PHP e adaptaram para funcionar em outras plataformas como Windows e Mac OS. É importante citar que o servidor interpreta o código e retorna a página HTML sem que o usuário final tenha conhecimento de qual sistema operacional está sendo executado no servidor.

Além dos desenvolvedores que trabalharam no desenvolvimento de sua própria versão do PHP, existem desenvolvedores que focaram no desenvolvimento de *frameworks* para a linguagem, isto é, desenvolveram um modelo de trabalho e ferramentas que podem ser utilizadas em diversos projetos. Normalmente, o desenvolvedor escolhe o *framework* a ser utilizado por suas características, visando o escopo do projeto e sua experiência. Dentre os *frameworks* mais conhecidos, podem ser citados:

1. Cake PHP: utiliza padrões de projeto conhecidos: Modelo-Visão-Controle (MVC, do inglês *Model-View-Controller*), *ActiveRecord*,

AssociationDataMapping e *FrontController*. Permite ao desenvolvedor criar aplicações robustas rapidamente.

2. *Zend Framework*: conhecido também como biblioteca de componentes, também oferece suporte para implementações *MVC*.

A empresa *Zend - The PHP Company*, responsável pelo *Zend Framework*, oferece certificação para PHP e para o próprio *framework*. Porém, segundo *Zend (2010)*, no Brasil existem apenas 142 profissionais certificados, o que nos permite afirmar que não existe exigência sobre as certificações em PHP nas empresas brasileiras.

1.2 AMBIENTE DE DESENVOLVIMENTO

Existem vários sistemas de software que permitem a edição de arquivos PHP. É possível que se desenvolva sistemas usando, desde o mais simples editor de texto (Bloco de Notas do Microsoft Windows, por exemplo) até ambientes de desenvolvimento (IDE) completas como o Eclipse, conforme *Milani (2010)*. Importante é escolher um IDE que garanta maior produtividade ao desenvolvedor.

Mas é necessário que, além de um IDE, o desenvolvedor tenha um servidor disponível para que possa visualizar e testar sua aplicação. É possível que esse servidor seja configurado nas plataformas Linux, Windows e Mac OS, sendo nativo apenas para o Linux. Para habilitá-los nesses sistemas operacionais, é necessário a instalação e execução de um servidor Web, que pode ser:

1. *Apache¹ HTTP Server*: desenvolvido em 1995 e mantido pela *The Apache Software Foundation*. É o mais popular entre os servidores Web, podendo ser instalado em plataformas Microsoft Windows, Linux e Mac OS.
2. *Nginx²*: desenvolvido em 2005 pelo russo Igor Sysoev. Está na versão 0.8.52 e pode ser instalado em Microsoft Windows e Linux.

¹ Apache HTTP Server: <http://www.apache.org/>

² Nginx: <http://www.nginx.org/>

3. *Lighttpd*³: desenvolvido em 2003 por Jan Kneschke. Atualmente, encontra-se na versão 1.4.28 e apenas pode ser instalado no Linux.

Essas ferramentas podem ser instaladas na própria máquina ou em um servidor dedicado, dependendo apenas da demanda e estrutura física disponível. Dentre as ferramentas citadas, segundo Ediger (2008), o *Apache* é o que mais consome recursos do servidor, porém é o mais conhecido, facilmente configurável e integra facilmente com seu ambiente. Ediger (2008), considera o *Lighttpd* como preferencial pelo motivo de utilizar de poucos recursos do servidor e indica o *Nginx* como promissor porque, apesar de novo, possui um conjunto de recursos comparável com o *Apache* e consome pouco recurso do servidor como o *Lighttpd*.

Após a definição do servidor é necessário efetuar a instalação do módulo PHP, responsável pela interpretação com códigos. Ainda, para o desenvolvimento de sistemas convencionais é necessário a instalação e configuração de um SGBD. São alguns exemplos compatíveis com PHP:

1. MySQL⁴: Desenvolvido na década de 90, foi projetado inicialmente para aplicações de pequeno e médio porte, mas hoje atende a aplicações de grande porte e com mais vantagens do que seus concorrentes, segundo Milani (2006).
2. PostgreSQL⁵: Também desenvolvido na década de 90, mais precisamente no ano de 1994, é de livre distribuição e código aberto e se destaca pela capacidade de armazenamento, segundo Leite (2007).
3. Microsoft SQL Server⁶: Banco de dados relacional desenvolvido em 1988 pela Microsoft e Sybase. Atualmente mantido apenas pela Microsoft e está na versão 2008. Uma desvantagem é que apenas pode ser instalado em servidores que executem sobre a plataforma Microsoft Windows.

³ *Lighttpd*: <http://www.lighttpd.net/>

⁴ MySQL: <http://www.mysql.com>

⁵ PostgreSQL: <http://www.postgresql.org/>

⁶ Microsoft SQL Server: <http://www.microsoft.com/sqlserver/>

4. Oracle⁷: desenvolvido no final dos anos 70 e muito utilizado em grandes corporações. É mantido pela Oracle *Corporation*.

Esses SGBDs são amplamente utilizados nos sistemas de software desenvolvidos com a linguagem PHP e são selecionados de acordo com a necessidade da corporação, quantidade de informação que será armazenada, escalabilidade e infraestrutura disponível.

1.3 PERFIL DO DESENVOLVEDOR

Um experiente desenvolvedor PHP não indica que seja bem remunerado. Segundo revista Info (2010), a remuneração de desenvolvedores PHP não ultrapassa R\$5.000,00, mais especificamente, varia de R\$2.500,00 a R\$5.000,00, dependendo do nível de conhecimento do desenvolvedor.

É possível verificar que, no portal Ceviu (2010), existem cadastradas 379 vagas de estágio ou emprego para desenvolvedor PHP em todo Brasil, distribuídas principalmente entre São Paulo (143 vagas), e Minas Gerais (140 vagas). São números relativamente pequenos se comparados a outras linguagens como Java e .NET da Microsoft.

⁷ Oracle: <http://www.oracle.com>

1.4 APLICAÇÕES

O PHP pode ser utilizado no desenvolvimento de sistemas de todos os portes, sendo mais utilizados para sistemas de pequeno e médio porte, como páginas pessoais, *blogsetc.*

O sistema Moodle, desenvolvido para atender instituições acadêmicas, é um sistema de médio a grande porte. O sistema é, segundo Moodle (2010), utilizado no Brasil por 3.325 portais incluindo a Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS) e da Universidade Federal do Espírito Santo (UFES).

Um exemplo de um sistema de grande porte, muito conhecido desenvolvido em PHP é o Facebook, um *site* de relacionamento na Internet. Para tal, a própria empresa desenvolveu uma extensão para PHP conhecida como *Hip Hop for PHP*, com o objetivo de reduzir o uso de processamento nos servidores Web em 50%, conforme Facebook (2010).

2. JAVA EE

A linguagem de programação Java é o resultado de um investimento *SunMicrosystems*, comprada recentemente pela *Oracle Corporation*. Quando, em 1991, financiou um projeto de pesquisa corporativa interna. Durante seu desenvolvimento, segundo Deitel e Deitel (2006), o projeto passou por dificuldades e quase colocou o projeto em risco. Por fim, o projeto seria cancelado se não fosse a explosão de popularidade da *World Wide Web* em 1993.

O Java *Enterprise Edition* (Java EE) define o padrão para desenvolver componentes baseados em aplicações empresariais multicamada, e provê serviços adicionais, ferramentas e Interface de Programação de Aplicativos (API – do inglês *Application Programming Interface*) para suportar o desenvolvimento simplificado das aplicações empresariais, segundo Somera (2006). Conforme Silva (2006), “Java EE é a tecnologia Java para aplicações corporativas disponíveis na internet ou não. É ideal para a construção de servidores de aplicação, integração de sistemas ou distribuição de serviços para terceiros.”.

Além da versão *Enterprise Edition*, a linguagem Java possui outras duas versões importantes, segundo Silva (2006):

1. Java *Standard Edition* (Java SE): tecnologia Java para computadores pessoais, *notebooks* e arquiteturas com poder de processamento e memória consideráveis. Enfoca o desenvolvimento de aplicação Cliente/Servidor sem suporte a tecnologias para Internet;
2. Java Micro Edition (Java ME): tecnologia Java para dispositivos móveis com limitações de memória ou processamento. São utilizadas para sistemas em celulares, *palmtops*, *pocket pcs*, *smartphones*, *javacards* e demais dispositivos móveis.

A principal diferença entre as versões é ambiente de execução. Tais edições são pacotes de sistemas de software que incluem os requisitos necessários de suporte a execução de programas Java, de acordo com Silva (2006). Os aplicativos Java são executados sobre uma máquina virtual (JVM, do inglês Java

Virtual Machine). Segundo Java (2010), “a JVM é um conjunto de programas de software que permite a execução de instruções – geralmente escritas em *bytecode* Java. Os JVMs estão disponíveis para todas as plataformas de software e hardware mais comuns.”

2.1 CARACTERÍSTICAS

O Java é uma linguagem orientada a objetos, robusta e segura. Segundo Allen e Bambara (2003), orientação a objetos é a metodologia utilizada para programar classes com base em objetos definidos e cooperativos. A orientação a objetos possibilita que haja criação de aplicações flexíveis e modulares, além de reutilização de códigos criados anteriormente, conforme Somera (2006).

A robustez da linguagem deu-se pela redução de possíveis erros em tempo de execução como inicialização automática de variáveis e por ser uma linguagem fortemente tipada, ou seja, as variáveis podem conter apenas um tipo de dados.

Uma importante característica do Java é a independência de plataforma, permitindo o desenvolvedor de escrever aplicativos uma única vez para executar em qualquer arquitetura. O motivo, segundo Somera (2006), é a geração de *bytecodes* a partir do código fonte. Os *bytecodes* são um conjunto de instruções, próximo a linguagem de máquina, mas não específicas de uma arquitetura. Na verdade, são instruções para a JVM.

Segundo Silva (2006), a plataforma Java EE utiliza um modelo de aplicação distribuída multicamada baseado em componentes que são executados em um servidor de aplicações. Segundo Bodoff *et al*(2002), a lógica da aplicação é dividida em componentes de acordo com sua função. Os diversos componentes podem ser instalados em diferentes equipamentos dependendo da camada a qual o componente pertence.

2.2 AMBIENTE DE DESENVOLVIMENTO

O desenvolvimento de sistemas de software utilizando a linguagem Java consiste em cinco fases, segundo Deitel e Deitel (2010):

1. Edição: O programa é escrito em um editor de texto – Bloco de Notas para plataforma Windows, por exemplo. Nesta fase o arquivo deve ser salvo com extensão `.java`.
2. Compilação: Fase responsável pela geração dos *bytecodes* a partir do código do sistema. O compilador armazena em disco um arquivo de extensão `.class` contendo os *bytecodes*.
3. Carga: O carregador de classe do Java, lê os arquivos `.class` e os carrega na memória principal.
4. Validação: verifica se os *bytecodes* carregados na memória são válidos e não violam restrições de segurança da linguagem.
5. Execução: A JVM lê os *bytecodes* da memória e compila para uma linguagem que o computador possa entender.

Os cinco processos são obrigatórios e necessários para desenvolvimento de um sistema. Contudo, essas etapas podem ser simplificadas quando utilizado um ambiente de desenvolvimento integrado (IDE, do inglês *Integrated Development Enviroment*). IDEs fornecem ferramentas que suportam o processo de desenvolvimento de software, incluindo editores, depuradores de código. As mais conhecidas são: Eclipse⁸, NetBeans⁹, JBuilder¹⁰, JCreator¹¹ etc.

O objetivo das IDEs é facilitar e agilizar o desenvolvimento do sistema de software. O Eclipse, por exemplo, permite que a aplicação seja organizada, escrita, compilada e executada em um mesmo ambiente. Porém, para colocar a aplicação disponível para acesso aos usuários é necessária a utilização de servidores, por exemplo, TomCat e JBoss dependendo do tipo de aplicação desenvolvida.

⁸ Eclipse: <http://www.eclipse.org>

⁹ NetBeans: <http://www.netbeans.org>

¹⁰ JBuilder: <http://www.codegear.com>

¹¹ JCreator: <http://www.jcreator.com>

O TomCat é um servidor Web que possui algumas características de um servidor de aplicação. O JBoss é um servidor de aplicação baseado na plataforma Java EE e desenvolvido utilizando a linguagem Java. Podemos citar outros servidores como IBM WebSphere da IBM, Borland Enterprise Sever, Oracle 9i Application Server etc.

2.3 PERFIL DO DESENVOLVEDOR

O desenvolvedor Java deve estar preparado para estudar e certificar-se. Pelo crescente mercado, as empresas procuram profissionais experientes e preferencialmente certificados. No *site* da Oracle (2010) é possível encontrar informações sobre certificações:

1. Java EE 5 *Web ComponentDeveloper*: certificação voltada para desenvolvedores que estão especializando em aplicações *JavaServerPages* e *Servlets*, utilizado para apresentar páginas Web e *Web Services*.
2. Java EE 5 *BusinessComponentDeveloper*: foco para desenvolvedores de componentes, utilizando *EnterpriseJavaBeans* (EJB), responsáveis pela concepção, desenvolvimento, testes, implantação e integração de aplicações.
3. Java EE 5 *Web ServicesDeveloper*: para desenvolvedores que criam sistemas *Web Services* utilizando componentes da tecnologia Java.
4. Java EE 5 *Enterprise Architect*: certificação voltada para arquitetos de software, responsáveis pela arquitetura e desenho de aplicações.

Por outro lado, a quantidade de vagas ofertadas pelo mercado brasileiro é acima da média. No Brasil, de acordo com o portal Ceviu (2010), 1480 vagas estão aguardando um desenvolvedor qualificado. O estado de São Paulo oferece 673 vagas, Rio de Janeiro oferece 216 vagas e Minas Gerais apenas 188. Estados como Paraná, Distrito Federal e Mato Grosso do Sul ofertam próximo de cem vagas.

Outro motivo para o contentamento dos desenvolvedores é a faixa salarial. A revista Info (2010) divulgou que os salários de desenvolvedor Java variam

de R\$2.500,00 a R\$6.000,00, apresentando uma real vantagem sobre a linguagem PHP.

2.4 APLICAÇÕES

“A plataforma Java EE garante a robustez necessária a nível de acesso, funcionalidade e estabilidade das soluções tecnológicas” (Silva, 2006). O Java suporta tanto sistemas de pequeno e médio porte, como aplicações pesadas e de grande porte. Alguns exemplos, de acordo com site da Oracle *Corporation* (2010), são:

1. Siebel *CustomerRelationshipManagement* (CRM): completa solução em gerenciamento de relacionamento com cliente. Utilizada no Brasil por L’Oréal Brasil Ltda, Sky Brasil Serviços Ltda, Itaotec, Azul Linhas Aéreas Brasileiras S.A. etc.
2. Primavera: O foco dos produtos Primavera da Oracle é ajudar empresas com muitos projetos a gerenciar todo o ciclo de vida do portfólio, incluindo projetos de todos os tamanhos. São empresas que utilizam a ferramenta: Construtora Norberto Odebrecht AS, Matec Engenharia e o Tribunal Regional Eleitoral de Santa Catarina – TRESC etc.

A adesão de grandes empresas a aplicações Java demonstra a estabilidade, robustez e qualidade da linguagem e da plataforma.

3. ESTUDO DE CASO / AVALIAÇÃO DE PRODUTIVIDADE

Com o conhecimento das linguagens PHP e Java, fica sempre a questão da melhor linguagem para desenvolvimento de sistemas Web. Em vista disso, este capítulo visa avaliar a produtividade no desenvolvimento de um sistema hipotético utilizando ambas as linguagens. Nesse estudo de caso, além de mensurar o tempo de desenvolvimento, são avaliados dificuldades, obstáculos etc que foram enfrentados no decorrer do desenvolvimento.

Este capítulo está organizado conforme a seguir. Na seção 3.1, é apresentado o sistema motivador, ou seja, o sistema que será desenvolvido utilizando as linguagens PHP e Java para avaliação de produtividade. Na seção 3.2, descreve-se as principais características comuns às implementações e avalia-se o desenvolvimento em cada linguagem. Na seção 3.3, faz-se uma análise sobre as linguagens.

3.1 SISTEMA MOTIVADOR

O sistema motivador **XMovies** é um sistema hipotético e convencional planejado e desenvolvido para esta monografia. O sistema, baseado no funcionamento de uma vídeo locadora, é formado por:

1. Área para criar, ler, atualizar e remover registros de clientes de filmes (CRUD, do inglês *Create, Retrieve, Update e Delete*);
2. Tela *mestre/detalhe*, ou seja, para cada registro *mestre* existem nenhum ou vários registros *detalhe*. Em banco de dados é um relacionamento 1:n.
3. Relatórios

Nas seções seguintes, esse sistema é desenvolvido em PHP e em Java de acordo com uma especificação única conforme pode ser vista no **Apêndice A**. Esse anexo é uma especificação de requisitos no modelo Praxis e tem o objetivo de esclarecer todos os requisitos do sistema hipotético: o que será desenvolvido, o que não será desenvolvido, regras de negócios etc.

3.2 IMPLEMENTAÇÃO

Nesta seção são apresentados os passos, etapas, ferramentas utilizadas para a construção do sistema XMovies nas duas linguagens. Iniciamos o processo configurando o ambiente, cliente e servidor, conforme a seguir.

Utilizamos uma máquina cliente com seguinte ambiente:

1. Windows 7
2. IDE Eclipse Helios
3. Interface de gerenciamento de SGBD MySQL HeidiSQL

Também configuramos uma máquina servidor com o seguinte ambiente:

1. Windows 7
2. Apache Tomcat versão 6.0.9 – Porta de comunicação 8080
3. Apache Http Server versão 2.2 – Porta de comunicação 80
4. SGBD MySQL versão 5.1.59 – Porta de comunicação 3306

A próxima etapa é a definição do modelo de dados que será o mesmo para o sistema em PHP e para Java EE. O sistema obedece a seguinte estrutura de dados, conforme FIG.1:

1. Atendente: mantém o registro de todos os funcionários da Dumont Locadora
2. Cliente: as informações dos clientes como nome, CPF, endereço estarão contidas nessa tabela
3. Filme: informações sobre os filmes disponíveis para locação.
4. Locação: representa cada vez que o cliente fizer uma locação de um ou mais filmes.
5. Locação Filmes: é a tabela que representa o relacionamento n:n entre as tabelas locação e filme.

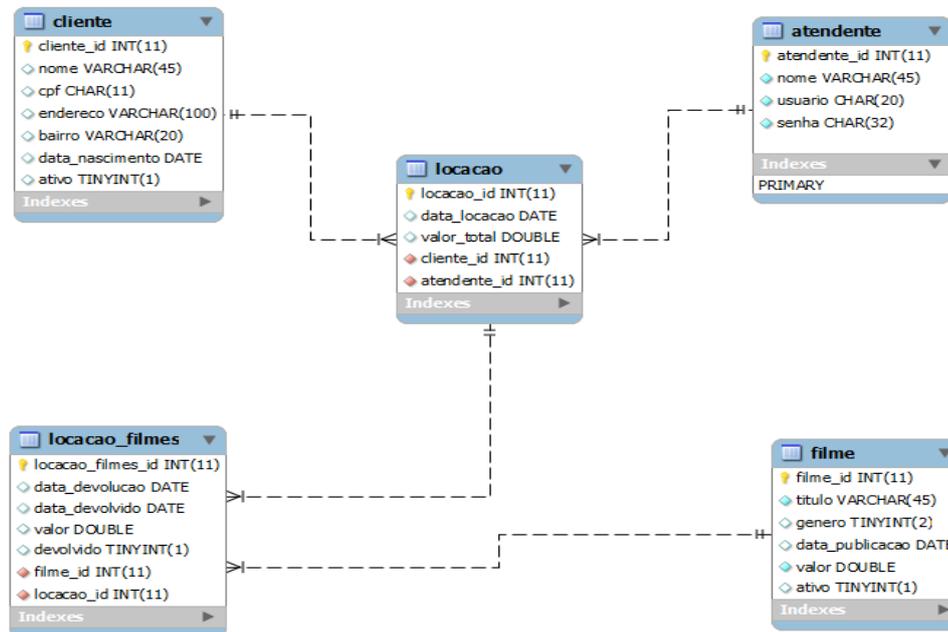


FIGURA 1 – Diagrama entidade-relacionamento (DER) utilizado no sistema XMovies.

Depois de definirmos a estrutura de dados, iniciamos o desenvolvimento do leiaute utilizando códigos HTML e *Cascading Styles Sheet* (CSS). Para essa tarefa iniciamos a mensuração de tempo para análise e foram utilizados 50 minutos.

Uma decisão tomada para atender a regra de negócio referente a segurança também foi comum às duas linguagens. Para que se tenha acesso completo ao sistema, o atendente deve estar autenticado no sistema e para garantir essa autenticação, utilizamos *cookies*.

O *cookie* é um conjunto de dados que são armazenados no computador do usuário e acessados pelo navegador Web. Um *cookie* é definido por uma chave e um valor.

Para o sistema utilizamos cinco *cookies*, conforme a seguir:

1. *Atendente_id*: é o código identificador do atendente na tabela do SGBD Atendente.
2. *Nome*: nome do atendente autenticado no sistema
3. *Usuário*: nome de usuário utilizado para acesso ao sistema
4. *Senha*: senha criptografada utilizada para acesso ao sistema

5. Checkup: esse *cookie* é o resultado de uma criptografia da associação de todos os demais *cookies* e a uma palavra chave presente no código do sistema. É utilizado para validar se houve alguma alteração nos valores dos demais *cookies*, uma vez que não existe decriptografia para a técnica de criptografia escolhida.

3.2.1 AVALIAÇÃO PHP

Após a definição de escopo e especificação de requisitos aprovadas, a primeira etapa para o desenvolvimento de um sistema, é a definição da estrutura de arquivos que será utilizada. Utilizamos a seguinte:

1. `_admin`: contém todos os arquivos que exigem que o Caixa ou Gerente estejam logados
2. `_classes`: contém apenas arquivos que representam as classes
3. `_config`: contém os arquivos de configuração `config.php` e `logado.php`
4. `_css`: contém arquivos de estilo
5. `_includes`: contém os arquivos que serão utilizados em mais de um arquivo

Essa definição de estrutura de arquivos é muito importante porque qualquer falha pode impactar todo o projeto e assim ocasionar atrasos na entrega do produto e, conseqüentemente, insatisfação do cliente.

Durante o desenvolvimento do sistema, utilizando apenas a linguagem de desenvolvimento PHP, foi observado todas as características como facilidades, obstáculos e barreiras enfrentadas, além da mensuração de tempo de desenvolvimento, conforme a seguir.

Para o desenvolvimento total do sistema XMovies foram gastos 12 horas e 17 minutos de trabalho, sem contar o tempo gasto para correção de *bugs* de natureza simples que totalizaram aproximadamente 15 minutos.

A principal dificuldade encontrada foi a definição da classe `Database`. Essa classe é estendida por todas as classes que necessitam de acesso ao SGBD, ou seja, todas as classes que utilizam do SGBD terão acesso aos métodos definidos na classe. A classe `Database` possui métodos genéricos para as principais ações como *SELECT*, *INSERT*, *UPDATE* e *DELETE*, como pode ser visto no código-fonte disponível no **Apêndice B**.

A linguagem PHP não requer controle do tipo de variável que está sendo definida. Qualquer variável definida durante o desenvolvimento do sistema pode receber diferentes tipos de dados em diferentes momentos. Essa é uma facilidade que permite maior agilidade durante o desenvolvimento do sistema.

Uma outra dificuldade enfrentada ocorreu com a manipulação de datas. A linguagem PHP não possui, funções nativas para o cálculo de diferença entre datas, conversões de data etc. Assim surgiu a necessidade de desenvolvermos uma classe específica para esses cálculos: `DataUtils`.

Apesar das dificuldades apresentadas, o desenvolvimento do sistema utilizando a linguagem PHP foi finalizado com sucesso.

3.2.2 AVALIAÇÃO JAVA EE

Utilizando a linguagem de desenvolvimento Java EE, o sistema XMovies ficou organizado da seguinte maneira:

1. Os *servlets* são responsáveis pelo contexto do sistema e possuem nome coeso com sua responsabilidade. Por exemplo, o *servlet* responsável pela parte de Clientes é o `ClienteServlet`.
2. As classes `Cliente`, `Filme` e `Locação` representam as tabelas `Cliente`, `Filme` e `Locação`, respectivamente, do SGBD.
3. Os arquivos que representam uma biblioteca de classes estão em um diretório denominado `lib`.

A primeira característica positiva, quando do desenvolvimento utilizando a linguagem Java EE é a facilidade de adaptarmos um *servlet* a partir de outro, desde

que os objetivos sejam semelhantes. Essa adaptação pode ser vista no código-fonte disponível no **Apêndice C**.

As semelhanças de ações entre `oservletsClienteServlet` e `FilmeServlet` permitiu agilizar o processo de desenvolvimento com reaproveitamento de código.

Uma característica, qualificada como negativa, sobre o Java EE ocorreu na inclusão de bibliotecas não nativas, para trabalharmos com o SGBD MySQL. O arquivo da biblioteca adicionada é `mysql-connector-java-5.1.13-bin.jar`.

Assim como o PHP, não existe uma classe nativa da linguagem que permita cálculos com datas. Novamente, é necessário desenvolver uma classe responsável por essas tarefas.

Apesar das dificuldades apresentadas durante o desenvolvimento do sistema, o XMovies foi finalizado com sucesso e atendendo aos requisitos especificados. Todo o processo de desenvolvimento foi concebido em 17 horas e 42 minutos. Novamente, sem contar com correções de *bugs* que são encontrados na etapa final do desenvolvimento do sistema.

3.3 ANÁLISE

O desenvolvimento de sistemas utilizando as linguagens PHP e Java EE é demasiadamente diferentes. As linguagens apresentam características semelhantes porém o corpo do código é muito diferente.

Como viram nas análises individuais das linguagens nas seções 3.1 e 3.2, a organização da estrutura dos arquivos é muito diferente. Contudo, aparentemente, percebe-se que o diretório é muito melhor organizado na estrutura do sistema desenvolvido em PHP.

No desenvolvimento em Java EE, foi desenvolvida uma classe específica `LayoutPortal`, para ser responsável pela apresentação das telas do sistema em suas diferentes situações. Isso facilita principalmente nos momentos de alteração no sistema, porque unificou-se as telas em apenas um arquivo, ao contrário no PHP que foram utilizados dois arquivos para representar o mesmo formulário em diferentes situações.

Quando se trata de trabalho com o SGBD, pode se dizer que o trabalho utilizando o PHP é mais simples. Em PHP, definimos um classe apenas para manipular os dados que são providos pelo SGBD. A classe `Database` possui métodos genéricos, seja para atualizar um registro ou obter uma lista completa de registros, e podem ser utilizados em todas as classes. Em Java EE, foi uma tarefa complicada e extensa. Foi necessário desenvolver, repetidamente, métodos semelhantes para as demais classes. Contudo, podia-se pensar em uma solução mais elegante e genérica, com uso de anotações e reflexão. Isso não foi realizado devido a um tempo considerável para desenvolvimento.

Além disso, no Java EE, é necessário acessar métodos distintos quando deseja-se fazer uma seleção de registros ou atualizar um registro no SGBD. No PHP é possível que o mesmo comando seja utilizado para seleção ou atualização, resultando em maior flexibilidade na estrutura do código.

Essas são as principais dificuldades encontradas durante o desenvolvimento do sistema XMovies além das pesquisas para utilização de classes e métodos nativos às linguagens.

3.4 AMEAÇAS À VALIDADE DA AVALIAÇÃO

Existem algumas características que podem representar ameaças à validade da avaliação. O autor deste estudo atua há sete anos no desenvolvimento de sistemas, sendo cinco anos em desenvolvimento utilizando a linguagem PHP. Durante um ano e seis meses desenvolveu sistemas utilizando a linguagem de programação Java EE.

4. CONCLUSÃO

A dificuldade inicial de escolher uma linguagem de desenvolvimento de sistemas Web atinge os desenvolvedores de sistemas e algumas empresas, principalmente aqueles que são novos ao mercado. O fator principal para essa dificuldade é a quantidade imensurável de linguagens de desenvolvimento existente e que aparecem a cada dia.

Assim, o objetivo deste estudo é facilitar essa escolha. Por se tratar de uma dificuldade que atinge, normalmente, novas empresas e novos desenvolvedores, escolheu-se duas linguagens de desenvolvimento de sistemas Web – PHP e Java EE – para que sejam avaliadas, levando em conta a produtividade do desenvolvimento e os benefícios de cada uma das linguagens durante todo o projeto.

Para fazer a avaliação de produtividade, efetuou-se o planejamento e o desenvolvimento de um sistema convencional. Durante o desenvolvimento, fez-se a mensuração do tempo gasto com a programação do sistema e também do tempo utilizado em pesquisas e testes específicos. Destacou-se, principalmente, as dificuldades enfrentadas durante o desenvolvimento, como também as facilidades e benefícios apresentados por cada linguagem.

O PHP, primeira linguagem utilizada neste estudo para o desenvolvimento do sistema convencional, saiu melhor no desenvolvimento completo do sistema. Isso envolveu o planejamento, as pesquisas, os testes e toda a programação do sistema. Porém, deparou-se com dificuldades em que suas soluções deveriam se tornar nativas a linguagem.

Como exemplo, pode-se citar a manipulação de datas. Foi necessário desenvolver uma classe que fosse responsável por operações como cálculo da diferença em dias entre datas, suas conversões etc. Outra dificuldade refere-se à definição de uma classe específica para as tarefas básicas relacionadas ao SGBD, assim como *SELECT*, *INSERT*, *UPDATE* e *DELETE*.

Entretanto, PHP é uma linguagem que atende às principais demandas de sistemas Web com mercado amplo e significativo. É uma solução de baixo custo para empresas que desejam entrar no mercado e aceita por grande parte dos desenvolvedores.

Por outro lado, Java EE teve uma programação mais eficaz apesar do tempo utilizado com pesquisas e testes. É uma linguagem mais organizada que o PHP e, por isso, conseguiu-se melhor desempenho durante o desenvolvimento do sistema.

Porém, assim como a linguagem PHP, enfrentou-se dificuldades referente a manipulação de datas. Na linguagem Java EE, existe uma classe nativa responsável por manipulação de datas, porém incompleta. Muitos desenvolvedores colaboradores tentaram, alguns ainda tentam, resolver esses problemas desenvolvendo classes próprias e contribuindo com a linguagem. O principal problema é que são diversas classes “filhas” da classe nativa e, normalmente, muito complexas para serem utilizadas em apenas um cálculo de diferença de datas, sendo mais viável desenvolver uma classe própria e que atenda a necessidade da demanda.

Uma característica negativa da linguagem Java EE é a necessidade de inclusão de arquivos externos contendo bibliotecas de classes e métodos não nativos à linguagem. Por exemplo, foi necessária a inserção de uma biblioteca externa para o acesso ao SGBD MySQL.

O sistema motivador permitiu apresentar as principais qualidades de cada linguagem, além de permitir a definição de uma linguagem ideal para esses sistemas convencionais.

Conforme já citamos, no processo de desenvolvimento de sistemas utilizando a linguagem de programação PHP foi utilizado 12 horas e 17 minutos. Em Java EE, o tempo foi superior, chegando a 17 horas e 42 minutos. Todavia, o tempo dedicado apenas à programação em Java EE foi menor que em PHP. Apenas 8 horas e 12 minutos foram utilizadas em programação do sistema em Java EE contra 10 horas e 38 minutos em PHP.

Essa diferença surgiu por, como pôde ser observado no item 3.4 (Ameaças à validade da Avaliação), o desenvolvedor responsável pela programação do sistema é mais experiente em PHP e para que fosse possível igualar o nível de conhecimento entre as linguagens decidiu-se considerar, com maior peso, o tempo de programação e efetuar mais de uma mensuração de tempo para concluir este estudo.

Assim, com base no estudo realizado, individualmente a cada linguagem, e em uma experiência clara e objetiva, decidiu-se que a linguagem Java EE é a mais

recomendada para desenvolvimento de sistemas convencionais para Web. Levou-se em conta a possibilidade de reaproveitamento, parcial ou completo, do código-fonte do sistema desenvolvido, bem como o tempo de programação em cada linguagem e, por fim, considerou-se de forma quase irrelevante, o tempo de pesquisa e testes, uma vez que esses serão únicos e poderão também ser reaproveitados.

REFERÊNCIAS

ALLEN, Paul R.; BAMBARA, Joseph J. **J2SE:Sun Certified Enterprise Architect for J2EE**. São Paulo: Elsevier Editora, 2003. 376 p.

CAKE PHP. Disponível em: <<http://cakephp.org/>>. Acesso em: 14 out. 2010.

CATTEL, Rick; INSCORE, Jim. **J2EE: Criando aplicações comerciais com a plataforma Java 2 Enterprise Edition**. São Paulo: Editora Campus, 2001. 298 p.

CONCEITOS e definições úteis. Java. Disponível em: <http://www.java.com/pt_BR/download/faq/helpful_concepts.xml>. Acesso em: 19 out. 2010.

DALL'OGGIO, Pablo. **PHP: programando com Orientação a Objetos**. 2. ed. São Paulo: Novatec Editora, 2009. 574p.

DEITEL, Harvey M.; DEITEL, Paul J. **C++: como programar**. 5. ed. São Paulo: Pearson Education do Brasil, 2006. 1163 p.

DEITEL, Harvey M.; DEITEL Paul J.; NIETO T. R.; MCPHIE, D. C. **PERL: Como Programar**. Trad. Carlos Arthur Lang Lisbôa. Porto Alegre: Brookman, 2002.

EDIGER, Brad. **Advanced Rails**. Sebastopol: O'Reilly, 2008.

FACEBOOK. Disponível em: <<http://www.facebook.com>>. Acesso em: 18 out. 2010.

LEITE, Mário. **Acessando banco de dados com ferramentas RAD: aplicações em Visual Basic**. Rio de Janeiro: Brasport, 2007.

MELO Jr, Cleuton Sampaio de. **Guia do Java Enterprise Edition 5: desenvolvendo aplicações corporativas**. Rio de Janeiro: Brasport, 2007.

MILANI, André. **Construindo aplicações Web com PHP e MySQL**. São Paulo: Novatec Editora, 2010. 336p.

MILANI, André. **MySQL**: Guia do programador. São Paulo: Novatec Editora, 2006. 336p.

NIEDERAUER, Juliano. **Desenvolvendo websites com PHP**: aprenda a criar websites dinâmicos e interativos com PHP e bancos de dados. São Paulo: Novatec Editora, 2004. 269p.

ORACLE Certification Program. Oracle. Disponível em: <http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=402&p_nl=JJAV&p_org_id=1001&lang=US>. Acesso em: 06 out. 2010.

PERFIL do Programador PHP. Disponível em: <<http://www.sauloarruda.eti.br/2010/03/perfil-do-programador-php/>>. Acesso em: 29 set. 2010.

PESQUISA de vagas. Ceviu. Disponível em: <<http://www.ceviu.com.br/vaga/?view=vgext>>. Acesso em: 18 out. 2010.

SALÁRIOS. Info. Disponível em: <<http://info.abril.com.br/professional/salarios/>>. Acesso em: 29 set. 2010.

SITES. Moodle. Disponível em: <<http://moodle.org/sites/>>. Acesso em: 18 out. 2010.

SCHILD, Herbert; HOLMES, James. **A Arte do Java**. São Paulo: Elsevier Editora, 2003. 376p.

SILVA, Luiz Augusto Matos da. **Plataforma Java J2EE**: a atual pedida no cenário corporativo. Manaus: UNINORTE. Disponível em: <http://www.luizmatos.eti.br/Artigos/Plataforma_Java.pdf>. (Aluno do Programa de Pós-Graduação em Redes de Computadores da União Educacional do Norte). Acesso em: 2 nov. 2010

SOMERA, Guilherme. **Treinamento Profissional em Java**: aprenda a programar nesta poderosa linguagem. São Paulo: Digerati Books, 2006. 128p.

5. APÊNDICE A – Especificação de Requisitos do projeto XMovies

Braxileiro Soluções em Informática Ltda

**Especificação dos Requisitos do Software
XMovies 1.0**

Autores: Luis Felipe Corrêa Pérez

Belo Horizonte

10 de novembro de 2010

Aprovação

Aprovamos a Especificação dos Requisitos 1.0 do projeto XMovies.

Luis Felipe Corrêa Pérez

10/11/10

Versões revisadas anteriores

Não aplicável

Especificação dos Requisitos do Software

Sumário

Especificação dos Requisitos do Software	36
1 Introdução	38
1.1 Objetivos deste documento.....	38
1.2 Escopo do produto.....	38
1.2.1 Nome do produto e de seus componentes principais.....	38
1.2.2 Missão do produto.....	38
1.2.3 Limites do produto.....	38
1.2.4 Benefícios do produto	38
1.3 Materiais de referência.....	39
1.4 Definições e siglas	39
1.5 Visão geral deste documento	39
2 Descrição geral do produto.....	40
2.1 Perspectiva do produto.....	40
2.1.1 Diagrama de contexto.....	40
2.1.2 Interfaces de usuário	40
2.1.3 Interfaces de hardware.....	40
2.1.4 Interfaces de software	41
2.1.5 Interfaces de comunicação	41
2.1.6 Restrições de memória	41
2.1.7 Modos de operação	41
2.1.8 Requisitos de adaptação ao ambiente	41
2.2 Funções do produto	41
2.3 Usuários e sistemas externos.....	42
2.3.1 Descrição.....	42
2.3.2 Características dos usuários.....	42
2.4 Restrições.....	42
2.5 Hipóteses de trabalho	42
2.6 Requisitos adiados	42

3	Requisitos específicos.....	43
3.1	Requisitos de interface externa	43
3.1.1	Interfaces de usuário	43
3.1.1.1	Interface de usuário Acesso ao sistema.....	43
3.1.1.2	Interface de usuário Gestão de Clientes	45
3.1.1.3	Interface de usuário Gestão de Filmes.....	47
3.1.1.4	Interface de usuário Locação	49
3.1.1.5	Interface de usuário Devolução	50
3.1.1.6	Interface de usuário Relatório Filmes mais locados	52
3.1.1.7	Interface de usuário Relatório Gênero mais locado	53
3.1.2	Interfaces de hardware.....	54
3.1.3	Interfaces de software	54
3.1.4	Interfaces de comunicação	54
3.2	Requisitos funcionais	55
3.2.1	Diagramas de casos de uso	55
3.2.1.1	Diagrama de casos de uso Pacotes de casos de uso	55
3.2.1.2	Diagrama de casos de uso Administração.....	55
3.2.1.3	Diagrama de casos de uso Operacional	55
3.2.2	Casos de uso	56
3.2.2.1	Caso de uso Gestão de filmes	56
3.2.2.2	Caso de uso Locação	56
3.2.2.3	Caso de uso Devolução de filmes.....	57
3.3	Requisitos não funcionais	58
3.3.1	Requisitos de desempenho.....	58
3.3.1.1	Requisito de desempenho Tempo de resposta para exibição da lista de filmes.....	58
3.3.2	Requisitos de dados persistentes	58
3.3.2.1	Diagrama de classes persistentes	58
3.3.2.2	Classes persistentes	58
3.3.2.3	Propriedades das classes persistentes.....	59
3.3.3	Restrições ao desenho	59
3.3.4	Atributos da qualidade	59
3.3.5	Outros requisitos	59
4	Informação de suporte.....	59

Introdução

Objetivos deste documento

Descrever e especificar as necessidades da DumontLocadora Ltda. que devem ser atendidas pelo produto XMovies, bem como definir para os desenvolvedores o produto a ser feito.
Público-alvo: cliente, usuários e desenvolvedores do projeto XMovies

Escopo do produto

Nome do produto e de seus componentes principais

XMovies (Componente único)

Missão do produto

Apoio informatizado ao controle de locação de filmes da locadora Dumont Locadora Ltda

Limites do produto

O XMovies não realizará gestão financeira
O XMovies não permitirá buscas por filmes disponíveis

Benefícios do produto

Número de ordem	Benefício	Valor para o cliente
	Agilidade na locação dos filmes	Essencial
	Agilidade na devolução dos filmes	Essencial
	Economia de mão-de-obra	Desejável
	Melhor controle dos filmes atrasados	Desejável

Materiais de referência

Número de ordem	Tipo do material	Referência bibliográfica
	Entrevistas	Ata de entrevistas, que podem ser conseguidas com a secretaria da Monograph Ltda

Definições e siglas

Não aplicável

Visão geral deste documento

De acordo com o Padrão para Especificação de Requisitos de Software, ou seja:

Parte 2: Descrição geral do produto

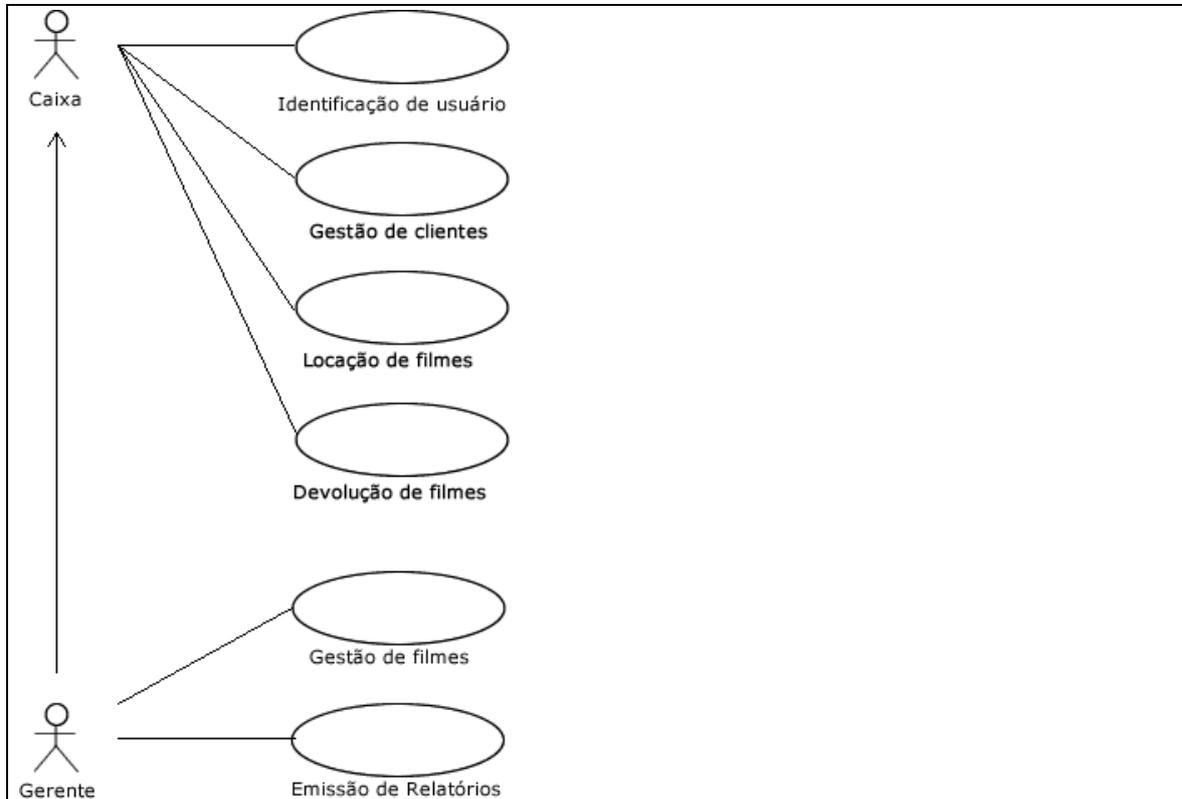
Parte 3: Requisitos específicos

Parte 4: Informação de suporte - listagens do Modelo de Análise

Descrição geral do produto

Perspectiva do produto

Diagrama de contexto



Interfaces de usuário

Número de ordem	Nome	Ator	Caso de uso	Descrição
1	Tela de cadastro de clientes	Caixa	Gestão de clientes	Interface on-line para cadastro de clientes
2	Tela de cadastro de filmes	Gerente	Gestão de filmes	Interface on-line para cadastro de filmes
3	Tela de locação de filmes	Caixa	Locação de filmes	Interface on-line para locação de filmes
4	Tela de devolução de filmes	Caixa	Devolução de filmes	Interface on-line para devolução de filmes

Interfaces de hardware

Não aplicável

Interfaces de software

Não aplicável

Interfaces de comunicação

Não aplicável

Restrições de memória

Número de ordem	Tipo de memória	Limites aplicáveis
1	HD	O produto deve ocupar no máximo 10MB (sem considerar a base de dados)
2	RAM	O produto deve executar em 128 MB RAM

Modos de operação

Número de ordem	Tipo de operação	Descrição da operação	Detalhes de operação
1	Interativa	MODO DE GESTÃO/VENDA	Modo de operação do XMovies , no qual o sistema está disponível para Gestão de clientes, filmes, locação e devolução de filmes.

Requisitos de adaptação ao ambiente

Não aplicável

Funções do produto

Número de ordem	Caso de uso	Descrição
1	Gestão de clientes	Cadastro, edição ou remoção de clientes que podem alugar vídeos
2	Gestão de filmes	Cadastro, edição ou remoção de filmes existentes em estoque
3	Locação de filmes	Locação de filmes por clientes
4	Devolução de filmes	Devolução de filmes por clientes

Usuários e sistemas externos

Descrição

Número de ordem	Ator	Definição
1	Caixa	Funcionário operador comercial de caixa.
2	Gerente	Funcionário responsável pela análise de relatórios, cadastro de filmes

Características dos usuários

Número de ordem	Ator	Frequência de uso	Nível de instrução	Proficiência na aplicação	Proficiência em informática
1	Caixa	Diário em horário comercial	1º Grau	Operacional	Aplicação
2	Gerente	Diário	2º Grau	Completa	Aplicação Windows XP

Restrições

Número de ordem	Restrição	Descrição
1	Ambiente	O ambiente operacional a ser utilizado é o Windows XP (ou compatível).
2	Expansibilidade	O produto deve ser desenvolvido levando-se em consideração que poderá ser expandido para mais de um caixa.
3	Segurança	O produto deverá restringir o acesso através de senhas individuais para cada usuário.

Hipóteses de trabalho

Número de ordem	Hipótese	De quem depende
1	Deve ser utilizado o sistema de gestão de bancos de dados MySQL	A Dumont Locadora deve adquirí-lo, instalá-lo e povoá-lo.

Requisitos adiados

Não aplicável

Requisitos específicos

Requisitos de interface externa

Interfaces de usuário

Interface de usuário Acesso ao sistema

Leiaute sugerido



The screenshot shows a login form for 'XMovies - Dumont Locadora'. It features two input fields: 'Usuário:' and 'Senha:'. Below the 'Senha:' field is a button labeled 'Acessar'. At the bottom of the form, there is a footer text: 'Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez'.

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

O comando *Acessar* acessa a área de administração e locação

Campos

Número	Nome	Descrição	Valores válidos	Formato	Tipo	Restrições
1	Usuário	Usuário cadastrado no sistema	Caracteres alfabéticos e espaços	Até 20 caracteres	Texto	Obrigatório / alterável.
2	Senha	Senha do usuário	Caracteres alfanuméricos	Até 8 caracteres	Texto	Obrigatório / alterável. A senha não deverá ser mostrada na tela, sendo substituída por um caractere especial.

Comandos

Número	Nome	Ação	Restrições
1	Acessar	Localiza o usuário e senha no sistema. Caso encontre permite o acesso ao sistema. Caso negativo, retorna a mesma tela para acesso	Sempre habilitado

Observações

<p>Não aplicável</p>

Interface de usuário Gestão de Clientes

Leiaute sugerido

XMovies - Dumont Locadora

[Clientes](#) | [Filmes](#) | [Locação](#) | [Devolução](#) | [Relatórios](#) | [Sair](#)

Cadastrar cliente

Nome

CPF

Endereço

Bairro

Data nascimento

Lista de clientes

Nome	CPF	Endereço	Bairro	Data de Nascimento	Ações
Luis Felipe	08054964676	Rua Nísio Batista de Oliveira	Novo São Lucas	28/11/1986	Editar Histórico Desativar cadastro
Caetano Motta	12345678902	Rua cobre, 200	Cruzeiro	20/01/1999	Editar Histórico Desativar cadastro

Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

Não aplicável

Campos

Número	Nome	Descrição	Valores válidos	Formato	Tipo	Restrições
1	Nome	Nome do cliente	Caracteres alfabéticos e espaços	Até 45 caracteres	Texto	Obrigatório / alterável.
2	CPF	Código de pessoa física	Caracteres numéricos	Até 11 caracteres	Texto	Obrigatório / Inalterável.
3	Endereço	Endereço completo de residência do cliente	Caracteres Alfanuméricos	Até 100 caracteres	Texto	Obrigatório/alterável
4	Bairro	Bairro de residência do cliente	Caracteres Alfanuméricos	Até 20 caracteres	Texto	Obrigatório/alterável
5	Data de Nascimento	Data de nascimento do cliente	Caracteres alfanuméricos	Até 10 caracteres	Texto	Obrigatório/alterável

Comandos

Número	Nome	Ação	Restrições
1	Cadastrar	Efetua o cadastro do cliente caso CPF ainda não exista no cadastro	Sempre habilitado

Observações

Não aplicável

Interface de usuário Gestão de Filmes

Leiaute sugerido

XMovies - Dumont Locadora

[Clientes](#) | [Filmes](#) | [Locação](#) | [Devolução](#) | [Relatórios](#) | [Sair](#)

Cadastrar filme

Título
 Gênero Selecione
 Valor Exemplo: 10,00

Lista de filme

Título	Valor	Data de Publicação	Ações
A Adolescente (La Joven)	R\$ 10.00	22/11/2010	Editar Desativar cadastro
A 20 Milhões de Milhas da Terra	R\$ 1.00	16/11/2010	Editar Desativar cadastro
Avatar	R\$ 5.00	15/11/2010	Editar Desativar cadastro
A Adolescente	R\$ 10.00	16/11/2010	Editar Desativar cadastro
Baby Looney Tunes Volume 1	R\$ 1.00	16/11/2010	Editar Desativar cadastro

Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

Não aplicável

Campos

Número	Nome	Descrição	Valores válidos	Formato	Tipo	Restrições
1	Título	Nome do filme	Caracteres alfabéticos e espaços	Até 45 caracteres	Texto	Obrigatório / alterável.
2	Gênero	Gênero que o filme pertence	Lista	-	ComboBox	Obrigatório / alterável.
3	Valor	Valor de Locação do filme	Entre 0 e 99,99	##,##	Ponto flutuante	Obrigatório/alterável

Comandos

Número	Nome	Ação	Restrições
1	Cadastrar	Efetua o cadastro do filme	Sempre habilitado

Observações

Não aplicável

Interface de usuário Locação

Leiaute sugerido

XMovies - Dumont Locadora

[Clientes](#) | [Filmes](#) | [Locação](#) | [Devolução](#) | [Relatórios](#) | [Sair](#)

Locação - Passo 1

Selecione o cliente

Locação - Passo 2

Selecione o(s) filme(s)

<input type="checkbox"/> A Adolescente	RS10,00	<input type="checkbox"/> A Adolescente (La Joven)	RS10,00
<input type="checkbox"/> Cabeça a Prêmio	RS23,12	<input type="checkbox"/> Da Vinci - O Crime Como Uma Arte	RS1,00
<input type="checkbox"/> E o Vento Levou 2	RS2,00	<input type="checkbox"/> Haifa	RS5,00

Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

Não aplicável

Campos

Número	Nome	Descrição	Valores válidos	Formato	Tipo	Restrições
1	Cliente	Nome do Cliente que está fazendo a locação	Lista	-	ComboBox	Obrigatório / inalterável.
2	Filmes	Filmes que o cliente está locando	Lista	-	CheckBox	Obrigatório / inalterável.

Comandos

Número	Nome	Ação	Restrições
1	Finalizar	Loca os filmes selecionados	Sempre habilitado

Observações

Não aplicável

Interface de usuário Devolução**Leiaute sugerido**

XMovies - Dumont Locadora

[Clientes](#) | [Filmes](#) | [Locação](#) | [Devolução](#) | [Relatórios](#) | [Sair](#)

Devolução - Passo 1

Selecione o cliente

Devolução - Passo 2

Selecione os filmes que serão devolvidos:

<input type="checkbox"/> El Cid	RS\$4,00	25/11/2010 (2 dia(s) para a data de devolução)
<input type="checkbox"/> G - Triangulo Amoroso	RS\$2,00	25/11/2010 (2 dia(s) para a data de devolução)

Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

Não aplicável

Campos

Número	Nome	Descrição	Valores válidos	Formato	Tipo	Restrições
1	Cliente	Nome do Cliente que está fazendo a devolução	Lista	-	ComboBox	Obrigatório / inalterável.
2	Filmes	Filmes que o cliente está devolvendo	Lista	-	CheckBox	Obrigatório / inalterável.

Comandos

Número	Nome	Ação	Restrições
1	Finalizar	Devolve os filmes selecionados	Sempre habilitado

Observações

Não aplicável

Interface de usuário Relatório Filmes mais locados

Leiaute sugerido

XMovies - Dumont Locadora

[Clientes](#) | [Filmes](#) | [Locação](#) | [Devolução](#) | [Relatórios](#) | [Sair](#)

Relatório Filmes Mais alugados

Posição	Qty	Título	Renda Total
1º	5	A 20 Milhões de Milhas da Terra	R\$ 5
2º	3	Baby Looney Tunes Volume 1	R\$ 3
3º	3	Avatar	R\$ 15
4º	2	G - Triangulo Amoroso	R\$ 4
5º	2	El Cid	R\$ 8
6º	2	A Adolescente (La Joven)	R\$ 20
7º	1	A Adolescente	R\$ 10
8º	1	Cabeça a Prêmio	R\$ 23,12
9º	1	Da Vinci - O Crime Como Uma Arte	R\$ 1
10º	1	Dark Water - Água Negra	R\$ 1

Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

Não aplicável

Campos

Não aplicável

Comandos

Não aplicável

Observações

Lista dos filmes mais locados

Interface de usuário Relatório Gênero mais locado**Leiaute sugerido**

XMovies - Dumont Locadora

[Clientes](#) | [Filmes](#) | [Locação](#) | [Devolução](#) | [Relatórios](#) | [Sair](#)

Relatório Gênero mais alugados

Posição	Qty	Gênero	Renda Total
1º	6	Ação	R\$ 15
2º	5	Animação	R\$ 45,12
3º	3	Chanchada	R\$ 15
4º	3	Catástrofe	R\$ 3
5º	2	Comédia dramática	R\$ 4
6º	2	Comédia romântica	R\$ 8

Sistema complementar da monografia "Avaliação de produtividade PHP x Java". Autor: Luis Felipe Corrêa Pérez

Diagrama de estados

Não aplicável

Relacionamentos com outras interfaces

Não aplicável

Campos

Não aplicável

Comandos

Não aplicável

Observações

Lista dos gêneros de filmes mais locados

Interfaces de hardware

Não aplicável

Interfaces de software

Não aplicável

Interfaces de comunicação

Não aplicável

Requisitos funcionais

Diagramas de casos de uso

Diagrama de casos de uso Pacotes de casos de uso

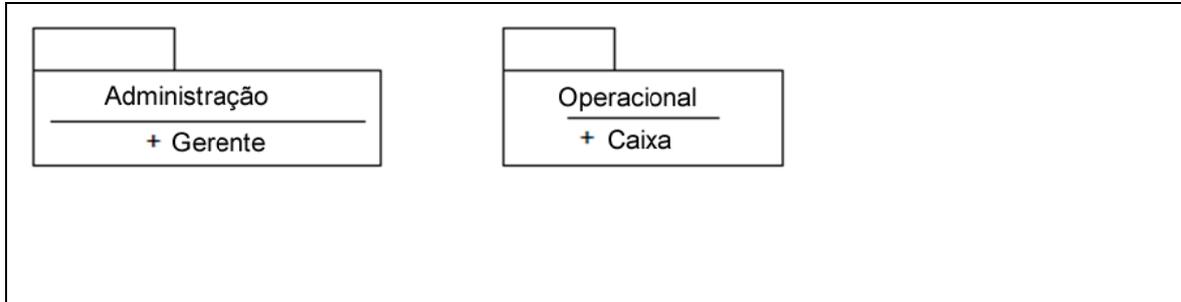


Diagrama de casos de uso Administração

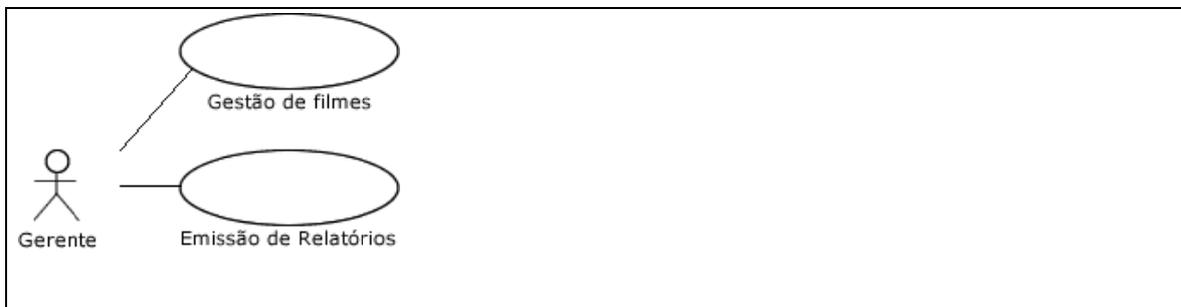
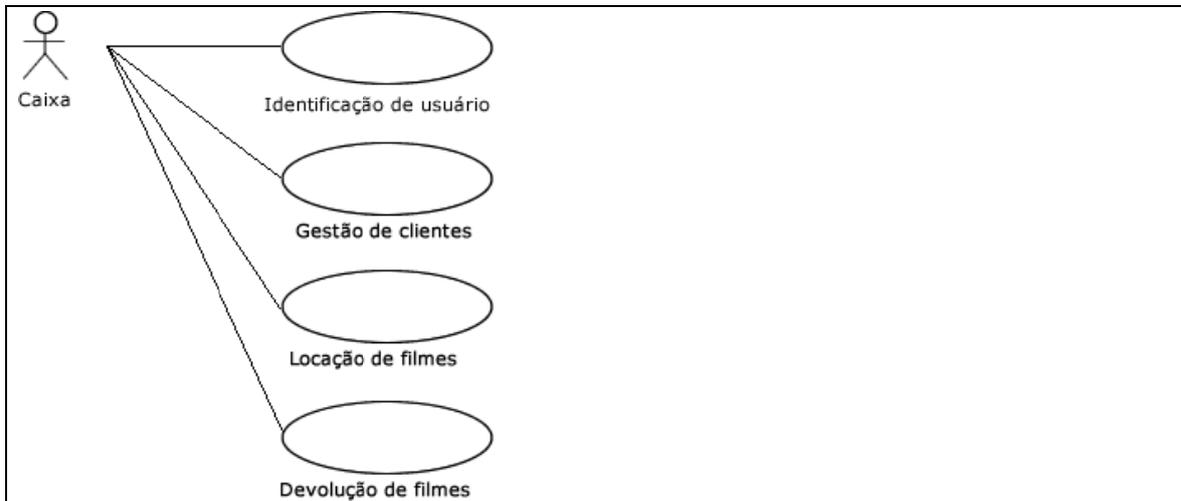


Diagrama de casos de uso Operacional



Casos de uso

Caso de uso Gestão de filmes

Precondições

O Gerente deve estar autenticado ao XMovies

Fluxo principal

- 1- O XMovies exibe a tela de cadastro
- 2- Gerente preenche os dados e aciona o comando *Cadastrar*
- 3- O XMovies efetua operação e retorna mensagem de sucesso ou erro

Subfluxos

Não aplicável

Fluxos alternativos

Fluxo alternativo Alteração de dados de filme

- 1- O Gerente seleciona um filme na lista
- 2- Altera os dados cadastrais
- 3- Salva e retorna a página principal de filmes

Fluxo alternativo Exclusão de Filme

- 1- O XMovies pede confirmação de exclusão
- 2- O Gerente confirma
- 3- O XMovies altera campo ativo = 0 do filme selecionado e limpa a tela

Caso de uso Locação

Precondições

O Caixa deve estar autenticado ao XMovies

Fluxo principal

- 1- O cliente seleciona os filmes que devem ser locados
- 2- O cliente informa seu nome ao Caixa
- 3- O Caixa seleciona o nome do cliente e clica em avançar
- 4- O XMovies lista os filmes existentes no sistema
- 5- O Caixa seleciona os filmes que estão sendo locados pelo cliente
- 6- O XMovies solicita a confirmação a locação
- 7- O Caixa confere e confirma a locação

Subfluxos

Não aplicável

Fluxos alternativos

Não aplicável

Caso de uso Devolução de filmes**Precondições**

O Caixa deve estar autenticado ao XMovies
 Cliente deve conter os filmes que deverão ser devolvidos

Fluxo principal

- 1- O XMovies exibe a lista de clientes que estão com filmes alugados
- 2- O Caixa seleciona o cliente que está efetuando a devolução
- 3- O XMovies exibe lista de filmes pendentes pelo cliente
- 4- O Caixa seleciona os filmes que estão sendo devolvidos e finaliza
- 5- O XMovies exibe mensagem confirmando a devolução

Subfluxos

Não aplicável

Fluxos alternativos

Não aplicável

Requisitos não funcionais

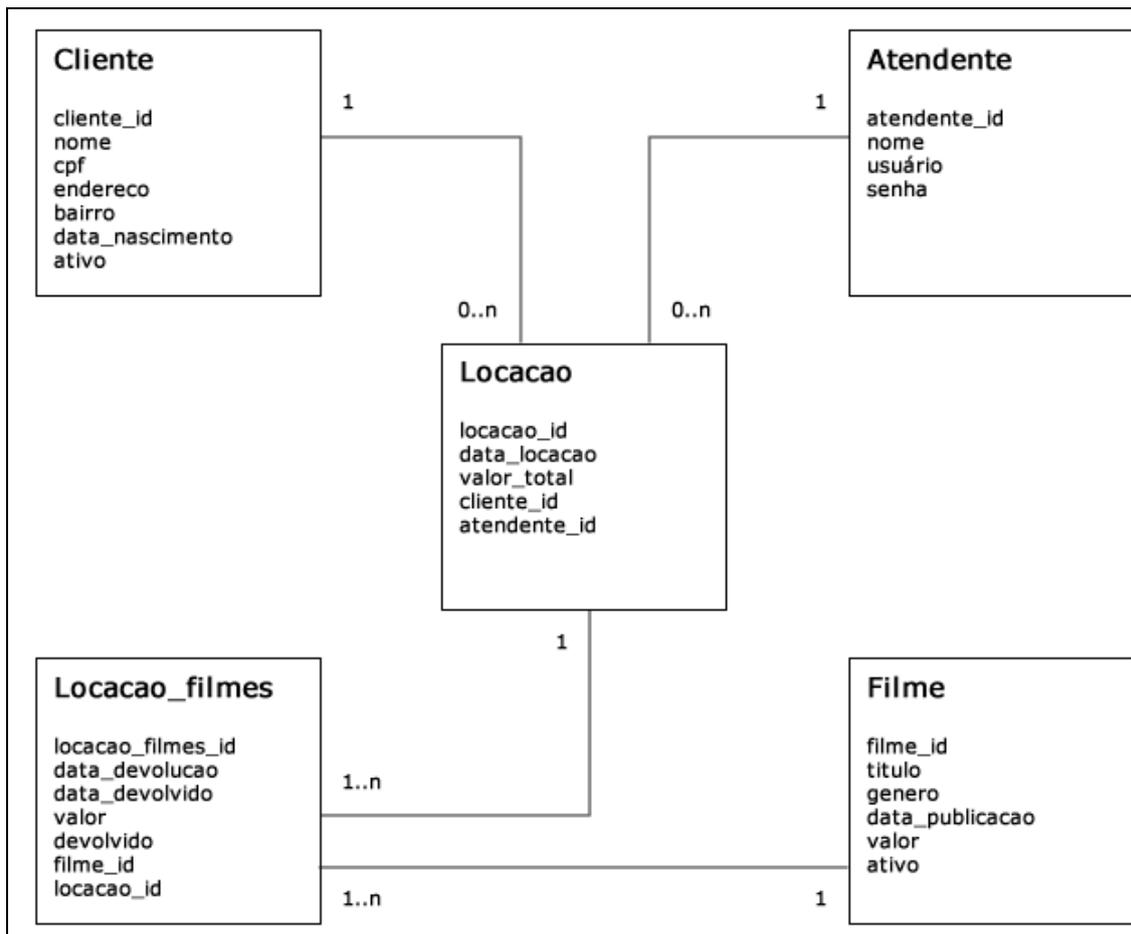
Requisitos de desempenho

Requisito de desempenho Tempo de resposta para exibição da lista de filmes

O tempo para realização de qualquer operação de listagem de filmes não pode ser superior a 10 segundos.

Requisitos de dados persistentes

Diagrama de classes persistentes



Classes persistentes

Número de ordem	Nome	Descrição
1	Cliente	Dados de clientes que fazem locação no sistema
2	Atendente	Dados do Caixa/Gerente que utilizam o sistema
3	Locacao	Dados gerais das locações dos filmes
4	Locacao_filmes	Dados gerais dos filmes locados por cliente
5	Filme	Dados dos filmes disponíveis para locação

Propriedades das classes persistentes

Número de ordem	Nome	Frequência de consulta (por dia)	Frequência de atualização (por dia)	Cardinalidade máxima	Restrições aplicáveis
1	Cliente	30	2	2000	Não excluir se houver pedidos de compra pendentes
2	Atendente	5	0,1	30	
3	Locacao	100	2	1000	
4	Locacao_filmes	100	4	10000	
5	Filme	100	4	10000	Não excluir se houver pedidos de compra pendentes

Restrições ao desenho

Não aplicável

Atributos da qualidade

Não aplicável

Outros requisitos

Não aplicável

Informação de suporte

Não aplicável

6. APÊNDICE B – Código fonte PHP

php/_config/logado.php

```

<?php
$objCheckup = new Checkup();
if(!isset($_COOKIE['atendente_id'],$_COOKIE['nome'],$_COOKIE['usuario'],$_C
COOKIE['senha'],$_COOKIE['checkup']))){
sair();
}
$dados = array( 'atendente_id'=> $_COOKIE['atendente_id'],
                'nome' => $_COOKIE['nome'],
                'usuario' => $_COOKIE['usuario'],
                'senha' => $_COOKIE['senha']
                );
if(!$objCheckup->validar($dados, $_COOKIE['checkup'])){

    sair();
}
function sair(){
    header('Location: /login_deslogar.php');
    exit('Sua sessão expirou.');
```

php/_config/config.php

```

<?php
//Cabecalho
header('Content-Type: text/html; charset=iso-8859-1');
//dados do cliente
define("CLIENTE_NOME","LocaFilmes Monografia");
//diretivas
error_reporting(E_ALL);
ini_set('display_errors','on');
define('DEBUGING', true);
//Chave de cookie
define('COOKIE_CHAVE','LocaFilmes2010Monografia');
define('COOKIE_DOMINIO','.' . $_SERVER['HTTP_HOST']);
define('COOKIE_PATH','/');
define('COOKIE_EXPIRA', time() + (3600 * 4 * 1)); //1 turno de 4 horas
define('COOKIE_EXPIRADO', time() - (3600 * 5 * 1)); //1 turno de 5 horas
atrás (garantia)
//Conexao banco de dados
define('MYSQL_LOCALHOST','localhost');
define('MYSQL_USUARIO','root');
define('MYSQL_SENHA','123456');
define('MYSQL_BANCO','locafilmes');
//configurações de pastas
define('PATH','E:/Desenvolvimento/PHP - Monografia/');
define('URL','http://www.xmovies.com.br/');
//configuracao de subpastas
define('URL_ADMIN',URL.'_admin/');
define('URL_CSS',URL.'_css/');
define('URL_JS',URL.'_script/');
define('URL_IMG',URL.'_imagens/');
define('URL_UPLOAD',URL.'_upload/');
define('URL_SWF',URL.'_swf/');
//separador de diretorios
define('DS',DIRECTORY_SEPARATOR);
//diretorios
define('DIR_CLASSES',PATH.'_classes'.DS);
define('DIR_CONFIG',PATH.'_config'.DS);
```

```

define('DIR_ADMIN',PATH.'_admin'.DS);
define('DIR_UPLOAD',PATH.'_upload'.DS);
//config do sistema
define('DIAS_LOCACAO', 3);

//Auto load de classes
function __autoload($classe) {
    if(is_file(DIR_CLASSES.$classe . '.php')){
        require_once DIR_CLASSES.$classe . '.php';
    }
    else
    exit("Classe não existente: " . $classe);
}
?>

```

php/_classes/Filme.php

```

<?php
class Filme extends Database{
    public $tabela = "filme";

    public function Filme(){
        parent::__construct();
    }

    public function getsIn($where_in){
        $sque = 'SELECT * FROM filme WHERE filme_id in ('. $where_in. ') ORDER BY
titulo ASC';
        return $this->QExec($sque);
    }

    function getsDisponiveisLocacao(){
        $sque = 'select f.*
        from filme f
        where filme_id not in (select filme_id
        from locacao_filmes
        where devolvido=0)
        order by titulo asc';
        return $this->QExec($sque);
    }

    function generoMaisAlugados(){
        $sque ='SELECT
        COUNT(*) AS qtd,
        f.genero,
        SUM(lf.valor) as renda
FROM
        locacao_filmes lf
        INNER JOIN filme f ON f.filme_id = lf.filme_id
        GROUP BY f.genero
        ORDER BY COUNT(*) DESC';
        return $this->QExec($sque);
    }

    function filmeMaisAlugados(){
        $sque = 'SELECT
COUNT(*) AS qtd,
        lf.filme_id,
f.titulo,

```

```

        SUM(lf.valor) as renda
FROM
    locacao_filmes lf
    INNER JOIN filme f ON f.filme_id = lf.filme_id
GROUP BY lf.filme_id
ORDER BY COUNT(*) DESC';
return $this->QExec($que);
}

function getsFilmesDevolucaoByCliente($cliente_id){
    $que = 'select f.*,locacao_filmes_id,data_devolucao, c.cliente_id
    from filme f
    inner join locacao_filmes lf on lf.filme_id = f.filme_id
    inner join locacao l on l.locacao_id = lf.locacao_id
    inner join cliente c on c.cliente_id = l.cliente_id
    where c.cliente_id = "'.$cliente_id.'"
    and lf.devolvido = 0';

    return $this->QExec($que);
}
}

```

php/_classes/Data.php

```

<?
class Data{
static function formataDataPTtoEN($data, $separador="-"){
    $data_formatada = @explode("/", $data);
    return
    $data_formatada[2].$separador.$data_formatada[1].$separador.$data_formatada
    [0];
}

    static function formataDataENTOPT($data, $separador="/"){
        if(strpos($data, "-")){
            $data_formatada = @explode("-", $data);
            if($data_formatada[2]=="00"){
                return "";
            }
            return
            $data_formatada[2].$separador.$data_formatada[1].$separador.$data_formatada
            [0];
        }
        return "";
    }

    static function validarData($data){
        if(strpos($data, "/")){
            $data = explode("/", $data);
            if(checkdate($data[1], $data[0], $data[2])){
                return true;
            }
            return false;
        }
        elseif(strpos($data, "-")){
            $data = explode("-", $data);
            if(checkdate($data[1], $data[2], $data[0])){
                return true;
            }
            return false;
        }
    }
}

```

```

    }
    else{
        return false;
    }
}

static function diferenca($data1, $data2){
    $data1 = self::strtotime($data1);
    $data2 = self::strtotime($data2);

    $diferenca = $data1 - $data2;

return (int)floor( $diferenca / (60 * 60 * 24));
}

static function strtotime($data){
    $data_ = explode('-', $data);
return mktime(0,0,0,$data_[1], $data_[2], $data_[0]);
}
}

```

php/_classes/Conexao.php

```

<?
class Conexao{
    private $conexao = NULL;
    private $bd = NULL;

    function Conexao(){
    $this->conexao = mysql_connect(MYSQL_LOCALHOST, MYSQL_USUARIO, MYSQL_SENHA)
or die("não foi possível conectar ao banco de dados.");
    $this->bd = mysql_select_db(MYSQL_BANCO) or die("Banco de dados
escolhido não existe no servidor.");
}

    function __destruct(){
        //@mysql_close($this->conexao);
    }

    function getConexao(){
        return $this->conexao;
    }

    function fechaConexao(){
        //@mysql_close($this->conexao);
    }
}
?>

```

php/_classes/Genero.php

```

<?php
class Genero{
    private $valores = null;

    public function Genero(){

$valores = array();
        $valores[1] = 'Ação';
        $valores[2] = 'Animação';
        $valores[3] = 'Aventura';
        $valores[4] = 'Chanchada';
        $valores[5] = 'Catástrofe';
        $valores[6] = 'Comédia';
        $valores[7] = 'Comédia romântica';
        $valores[8] = 'Comédia dramática';
        $valores[9] = 'Comédia de ação';
        $valores[10] = 'Cult';
        $valores[11] = 'Documentário';
        $valores[12] = 'Drama';

        asort($valores);

        $this->valores = $valores;
    }

    function get($id){
        if(isset($this->valores[$id]))
            return $this->valores[$id];
        else
            return false;
    }

    function getSelect($nome='genero', $selecionado = ''){
        $select = '<select name="'. $nome. '" id="'. $nome. '">';
        $select .= '<option value="">Selecione</option>';
        for($i=1; $i <= count($this->valores); $i++){
            $select .= '<option value="'. $i. '" ' . ($selecionado != '' && $i ==
$selecionado ? 'selected="selected"' : ''). '>'. $this-
>valores[$i]. '</option>';
        }

        $select .= '</select>';

        return $select;
    }
}

```

php/_classes/Locacao.php

```
<?php
class Locacao extends Database{
    public $tabela = "locacao";

    public function Locacao(){
        parent::__construct();
    }
}
```

php/_classes/Database.php

```
<?
class Database{

    private $objConexao;
    public $tabela = 'NOME_DA_TABELA';

function Database(){
    $this->objConexao = new Conexao();
}

function inserir($dados){
    $campo = "";
    $valor = "";
    $keys = array_keys($dados);
    $tamanho = sizeof($keys);
    for($i=0; $i<$tamanho; $i++){
        if($keys[$i]=="senha"){
            $campo .= ($i==0 ? "" : ",") . $keys[$i];
            $valor .= ($i==0 ? "" : ",") . "MD5('" . $dados[$keys[$i]] . "')";
        }
        else{
            $campo .= ($i==0 ? "" : ",") . $keys[$i];
            $valor .= ($i==0 ? "" : ",") . $dados[$keys[$i]];
        }
    }
    $que = 'INSERT INTO ' . $this->tabela . ' (' . $campo . ')VALUES(' . $valor . ')';
    return $this->QExec($que);
}

function remover($id){
    $que = 'UPDATE ' . $this->tabela . ' SET ativo=0 WHERE ' . $this->tabela . '_id = ' . $id;
    $exec = $this->QExec($que);
    if(mysql_affected_rows() > 0)
        return true;
    else
        return false;
}

function excluir($id){
    $que = 'DELETE FROM ' . $this->tabela . ' WHERE ' . $this->tabela . '_id = ' . $id;
    return $this->QExec($que);
}

function editar($id, $dados){
    $campo = "";
```

```

$valor = "";
$keys = array_keys($dados);
$tamanho = sizeof($keys);
for($i=0; $i<$tamanho; $i++){
    if($keys[$i]=="senha"){
        $campo .= ($i==0 ? "" : ",") . $keys[$i] . "=";
        $valor = "MD5('" . $dados[$keys[$i]] . "')";
        $campo .= $valor;
    }
    else{
        $campo .= ($i==0 ? "" : ",") . $keys[$i] . "=";
        $valor = "'" . $dados[$keys[$i]] . "'";
        $campo .= $valor;
    }
}
$sque = 'UPDATE ' . $this->tabela . ' SET ' . $campo . ' WHERE ' . $this->tabela . '_id = ' . $id;
return $this->QExec($sque);
}

function get($where){
    $clausula_where = '';
    if(is_array($where)){
        if(count($where)>0){
            $clausula_where = 'WHERE ';

            $keys = array_keys($where);
            $tamanho = sizeof($keys);
            for($i=0; $i<$tamanho; $i++){
                $clausula_where .= ($i==0 ? "" : " AND ") . $keys[$i] . "=";
                $valor = "'" . $where[$keys[$i]] . "'";
                $clausula_where .= $valor;
            }
        }
    }
    else{
        die('Necessita um array');
    }
}

$sque = 'SELECT * FROM ' . $this->tabela . ' ' . $clausula_where . ' LIMIT 1';
$exec = $this->QExec($sque);
return (mysql_num_rows($exec)>0) ? mysql_fetch_assoc($exec) : false;
}

function gets($where=array(), $inicio=0, $limite=0, $order=array(), $operador = "AND"){
    $clausula_where = '';

    if(count($where)>0){
        $clausula_where = 'WHERE ';

        $keys = array_keys($where);
        $tamanho = sizeof($keys);
        for($i=0; $i<$tamanho; $i++){
            $clausula_where .= ($i==0 ? "" : ' ' . $operador . ' ' ) . $keys[$i] . "=";
            $valor = "'" . $where[$keys[$i]] . "'";
            $clausula_where .= $valor;
        }
    }
}

```

```

    $clausula_order = '';
    if(count($order)>0){
        $clausula_order = 'ORDER BY ';

        $keys = array_keys($order);
        $tamanho = sizeof($keys);
        for($i=0; $i<$tamanho; $i++){
            $clausula_order .= ($i==0 ? "" : " , " ).$keys[$i]." ".$order[$keys[$i]];
        }
    }

    $limit = ($limite == 0)?'':'LIMIT '.$inicio.', '.$limite;
    $que = 'SELECT * FROM '. $this->tabela .' ' . $clausula_where . ' ' .
    $clausula_order . ' ' . $limit;
    return $this->QExec($que);
}

function QExec($que){
    $exec = mysql_query($que, $this->objConexao->getConexao()) or (DEBUGING
? die(mysql_error()):'');
    return $exec;
}
}
?>

```

php/_classes/Checkup.php

```

<?php
class Checkup{
    //Construtor
    function Checkup(){
    }

    //cria hash md5 com dados do usuario
    function criar($dados){
        $checkup = '';

        foreach($dados as $chave => $valor){
            $checkup .= $chave.'='.$valor;
        }

        return md5($checkup.COOKIE_CHAVE);
    }

    function validar($dados, $checkup){
        $novo_checkup = $this->criar($dados);
        if($novo_checkup == $checkup){
            return true;
        }
        return false;
    }
}
?>

```

php/_classes/Cliente.php

```

<?php
class Cliente extends Database{
    public $tabela = "cliente";

    public function Cliente(){
        parent::__construct();
    }

    function getsClienteDevolucao(){
        $que ='select c.cliente_id, c.nome
            from cliente c
            inner join locacao l on l.cliente_id = c.cliente_id
            inner join locacao_filmes lf on lf.locacao_id = l.locacao_id
            inner join filme f on f.filme_id = lf.filme_id
            where lf.devolvido = 0
            and c.ativo = 1
            group by c.nome
            order by nome asc';

        return $this->QExec($que);
    }
}

```

php/_classes/Atendente.php

```
<?php
class Atendente extends Database{
    public $tabela = "atendente";

    public function Atendente(){
        parent::__construct();
    }
}
```

php/_classes/LocacaoFilmes.php

```
<?php
class LocacaoFilmes extends Database{
    public $tabela = "locacao_filmes";

    public function LocacaoFilmes(){
        parent::__construct();
    }

    public function getsByLocacaoId($locacao_id){
        $sque = 'SELECT * FROM locacao_filmes lf INNER JOIN filme f ON
lf.filme_id = f.filme_id WHERE lf.locacao_id = "'. $locacao_id. "'';
        return $this->QExec($sque);
    }
}
```

php/_classes/Util.php

```
<?php
/*****
Arquivo util.php
Classe PHP para auxílio da programação
Criado por: Luis Felipe Corrêa Pérez
Empresa: Dataminas
*****/
class Util{

    public static function removerAcentos($string)
    {
        $a = array(
            "/[ÃÃÃÃÃ]/" => "A",
            "/[âââââ]/" => "a",
            "/[ÊÊÊÊÊ]/" => "E",
            "/[êêêêê]/" => "e",
            "/[ÎÎÎÎÎ]/" => "I",
            "/[îîîîî]/" => "i",
            "/[ÔÔÔÔÔ]/" => "O",
            "/[ôôôôô]/" => "o",
            "/[ÛÛÛÛÛ]/" => "U",
            "/[úúúúú]/" => "u"
        );
        return preg_replace(array_keys($a), array_values($a), $string);
    }

    public static function tornarAmigavel($s)
    {
        $s = ereg_replace("[ÃÃÃÃÃâââââªä]", "a", html_entity_decode($s));
        $s = ereg_replace("[ÊÊÊÊÊêêêêê]", "e", $s);
        $s = ereg_replace("[ôôôôôö°öööööö]", "o", $s);
    }
}
```

```

    $s = ereg_replace("[úûüüÛÛÛÛ]", "u", $s);
    $s = ereg_replace("[íîïîï]", "i", $s);
    $s = str_replace("ç", "c", $s);
    $s = str_replace("Ç", "C", $s);
    $s = str_replace("$", "s", $s);
    $s = ereg_replace(" ", "-", $s);
//    $s = ereg_replace(".", "-", $s);
    $s = ereg_replace("[\\(\\)]", "-", $s);
    $s = ereg_replace("&", "e", $s);
    $s = strtolower($s);

    while(ereg("--", $s)==TRUE)
        $s = str_replace("--", "-", $s);

    if (substr($s,-1)=="-")
        $s=substr($s,0,-1);
    $s = ereg_replace("[^a-z0-9_-]", "", $s);
    if(strlen($s)==0)
        return 'x';
    else
        return $s;
    return $select;
}

public static function formatarTelefone($numero){
    $aux = ereg_replace("[^0-9]", '', $numero);
    if(strlen($aux) == 10){
        $area = substr($aux,0,2);
    $prefixo= substr($aux,2,4);
    $linha = substr($aux,6,4);
    }
    elseif(strlen($aux) == 9){
        $area = substr($aux,0,2);
    $prefixo= substr($aux,2,3);
    $linha = substr($aux,5,4);
    }else
        return $numero;

    $format = "(".$area.") ".$prefixo."-".$linha;
    return($format);
}

    public static function unhtmlentities($cadena){
    $cadena = preg_replace('~&#x([0-9a-f]+);~ei', 'chr(hexdec("\\1"))',
    $cadena);
    $cadena = preg_replace('~&#([0-9]+);~e', 'chr(\\1)', $cadena);
    $trans_tbl = get_html_translation_table(HTML_ENTITIES);
    $trans_tbl = array_flip($trans_tbl);
    return strtr($cadena, $trans_tbl);
    }

    public static function validarValor($valor){
        if(preg_match('/^[0-9]+(,[0-9]{2})?$/', $valor)){
            return true;
        }
        return false;
    }
}

public static function formatarValorPTtoEN($valor){
    $count = substr_count($valor, ',') + substr_count($valor, '.');

```

```

        if($count > 1)
            return false;

        $valor = str_replace(',', '.', $valor);

        return $valor;
    }

    public static function formatarValorENTOtoPT($valor){
        $count = substr_count($valor, ',') + substr_count($valor, '.');

        if($count > 1)
            return false;

        $valor = str_replace('.', ',', $valor);

        return $valor;
    }
}
?>

```

php/index.php

```

<?php
//inclusao do arquivo de configuracao
include_once('_config/config.php');
//inclusao do topo
include(PATH . '_includes/topo.php');
if(isset($_COOKIE['atendente_id'], $_COOKIE['nome'], $_COOKIE['usuario'],
$_COOKIE['senha'], $_COOKIE['checkup']))
header("Location: _admin/");
//verifica se tem alguma mensagem de erro
if(isset($_GET['erro'])){
    echo '<div>'.$_GET['erro'].'</div>';
}
//tela de acesso
echo '
<form action="/login_validar.php" method="post">
<table>
<tr>
<td>Usuário:</td>
<td><input type="text" name="usuario" maxlength="20" /></td>
</tr>

<tr>
<td>Senha:</td>
<td><input type="password" name="senha" maxlength="10"></td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="Acessar"></td>
</tr>
</table>';

//inclusao do rodape
include('_includes/rodape.php');
?>

```

php/_includes/rodape.php

```

<?php echo '</div>
<div id="rodape">
    Sistema complementar da monografia "Avaliação de produtividade PHP x
Java". Autor: Luis Felipe Corrêa Pérez
</div>
</body>
</html>';
?>

```

php/_includes/topo.php

```

<?php echo '<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>'.CLIENTE_NOME.'</title>
<link rel="stylesheet" href="'.URL_CSS.'estilo.css" />
</head>
<body>
<div id="topo">
<h1>XMovies - Dumont Locadora</h1>
</div>
    '.(isset($_COOKIE['checkup']))?
    '<div id="menu">
<a href="'.URL_ADMIN.'cliente_index.php">Clientes</a> |
<a href="'.URL_ADMIN.'filme_index.php">Filmes</a> |
<a href="'.URL_ADMIN.'locacao_index.php">Locação</a> |
<a href="'.URL_ADMIN.'devolucao_index.php">Devolução</a> |
<a href="'.URL_ADMIN.'relatorios_index.php">Relatórios</a> |
<a href="'.URL.'login_deslogar.php">Sair</a>
</div>
    ':'')
    .'<div id="conteudo">';

```

php/login_validar.php

```

<?php
//inclusao do arquivo de configuracao
include_once('_config/config.php');
//erro padrao
$erro = 'Acesso inválido';
//verifica se os parametros existem
if(isset($_POST['usuario'], $_POST['senha'])){
    $usuario = addslashes($_POST['usuario']);
    $senha = md5(addslashes($_POST['senha']));

    //valida o acesso do atendente
    $objAtendente = new Atendente();
    $atendente = $objAtendente->get(array('usuario' => $usuario, 'senha' =>
$senha));

    //se o atendente existe
    if($atendente !== false){
        $objCheckup = new Checkup();
        $checkup = $objCheckup->criar(array('atendente_id'=>
$atendente['atendente_id'],
            'nome' => $atendente['nome'],
            'usuario' => $atendente['usuario'],
            'senha' => $atendente['senha']
        )
    );

        setcookie('atendente_id', $atendente['atendente_id'], COOKIE_EXPIRA,
COOKIE_PATH,COOKIE_DOMINIO);
        setcookie('nome', $atendente['nome'], COOKIE_EXPIRA,
COOKIE_PATH,COOKIE_DOMINIO);
        setcookie('usuario', $atendente['usuario'], COOKIE_EXPIRA,
COOKIE_PATH,COOKIE_DOMINIO);
        setcookie('senha', $atendente['senha'], COOKIE_EXPIRA,
COOKIE_PATH,COOKIE_DOMINIO);
        setcookie('checkup', $checkup, COOKIE_EXPIRA,
COOKIE_PATH,COOKIE_DOMINIO);

        header('Location: /_admin');
        exit;
    }
    else{
//erro caso nao exista
        $erro = 'Usuário ou senha inválido';
    }
}
//envia mensagem de erro para a index
header("Location: /index.php?erro=" . $erro);
?>

```

php/_admin/devolucao.3.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
$objLocacaoFilmes = new LocacaoFilmes();
$msg = 'Erro. Tente novamente';
if(isset($_POST['filmes'])){
    if(count($_POST['filmes']) > 0){
        $cont = 0;
        $contFail = 0;
        foreach($_POST['filmes'] as $key => $value){
            $objLocacaoFilmes->editar( $value,
                array( 'devolvido' => '1',
                    'data_devolvido' => date('Y-m-d')
                )
            );
            if(mysql_affected_rows() > 0)
                $cont++;
        }
    }
    else{
        $contFail++;
    }

    $msg = $cont . ' filmes devolvidos com sucesso' . ($contFail > 0 ? ' e
    '.$contFail.' sem sucesso:');
}
else{
    $msg = 'Erro: Nenhum filme foi selecionado';
}
}
header('Location: devolucao_index.php?msg=' . $msg);
?>

```

php/_admin/locacao.3.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
if(!isset($_POST['cliente_id']) || $_POST['cliente_id'] == ''){
    header('Location: locacao_index.php?msg=Selecione um cliente');
}
elseif(!isset($_POST['filmes']) || $_POST['filmes'] == '' ||
count($_POST['filmes']) < 1){
    header('Location: locacao_index.php?msg=Nenhum filme foi selecionado');
}
echo '<h2>Locação - Confirmação</h2>';
<div class="espacado">
<form action="locacao.4.php" method="POST">
<input type="hidden" name="cliente_id" value="".$_POST['cliente_id'].'" />
<table>
<tr>
<td valign="top">Selecione o(s) filme(s)</td>
<td>';

        $objFilme = new Filme();
        $filmes = $objFilme->getsIn(join(',', $_POST['filmes']));
        $tamanho = mysql_num_rows($filmes);
        if($tamanho > 0){

```

```

        echo '<table border="1" cellspacing="0"
cellpadding="5">'.PHP_EOL.'';
echo '<tr>'.PHP_EOL.'';
$cont = 1;
    $total = 0;
    while($filme = mysql_fetch_assoc($filmes)){
        $bgcolor = ($cont % 2 == 0?'FFFFFF' : 'EFEFEF' );
        echo '<td bgcolor="#'.$bgcolor.'"><label><input type="checkbox"
checked="checked" name="filmes[]" value="'.$filme['filme_id'].'"
/>'.$filme['titulo'].'</label></td><td bgcolor="#'.$bgcolor.'"
align="right"><b>R$ '.
Util::formatarValorEntoPT($filme['valor']).'</b></td>'.PHP_EOL.'';
        $total += $filme['valor'];
        if($cont % 1 == 0 && $cont < $tamanho)
            echo '</tr>'.PHP_EOL.'<tr>';

        $cont++;
    }
    echo '</tr>';
echo '<tr bgcolor="#FF9999"><td>Total:</td><td><b>R$ '.
number_format($total,2,',','.').'</b></td></tr>';
    echo '</table>';
}
echo '</td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="Finalizar" /></td>
</tr>
</table>
</form>
</div>
';
include('../_includes/rodape.php');
?>

```

php/_admin/locacao_index.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
echo '<h2>Locação - Passo 1</h2>';
<div class="espacado">
<form action="locacao.2.php" method="POST">
'.(isset($_GET['msg']) ? $_GET['msg'] : '').'
<table>
<tr>
<td>Selecione o cliente</td>
<td>
<select name="cliente_id">
<option value=""></option>';

        $objCliente = new Cliente();
$clientes = $objCliente->gets(array('ativo' => 1), 0, 0, array('nome' =>
'ASC'));
while($cliente = mysql_fetch_assoc($clientes)){
    echo '<option
value="' . $cliente['cliente_id'] . '">' . $cliente['nome'] . '</option>';
    }

echo ' </select>
</td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="Prosseguir passo 2" /></td>
</tr>
</table>
</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/locacao.2.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
if(!isset($_POST['cliente_id']) || $_POST['cliente_id'] == ''){
header('Location: locacao_index.php?msg=Selecione um cliente');
}
echo '<h2>Locação - Passo 2</h2>';
<div class="espacado">
<form action="locacao.3.php" method="POST">
<input type="hidden" name="cliente_id" value="'.$_POST['cliente_id'].'" />
<table>
<tr>
<td valign="top">Selecione o(s) filme(s)</td>
<td>';

        $objFilme = new Filme();
        $filmes = $objFilme->getsDisponiveisLocacao();
        $tamanho = mysql_num_rows($filmes);
if($tamanho > 0){
        echo '<table border="1" cellspacing="0"
cellpadding="5">'.PHP_EOL.'';
echo '<tr>'.PHP_EOL.'';
        $cont = 1;
        while($filme = mysql_fetch_assoc($filmes)){
                $bgcolor = ($cont % 2 == 0?'FFFFFF':'EFEFEF');
                echo '<td bgcolor="#'.$bgcolor.'"><label><input type="checkbox"
name="filmes[]" value="'.$_POST['cliente_id'].'_'.$filme['filme_id'].'"
/>'. $filme['titulo'].'</label></td><td bgcolor="#'.$bgcolor.'"
align="right"><b> R$'.
Util::formatarValorEntoPT($filme['valor']).</b></td>'.PHP_EOL.'';
                if($cont % 3 == 0 && $cont < $tamanho)
                        echo '</tr>'.PHP_EOL.'<tr>';

                $cont++;
        }
        echo '</tr>';
echo '</table>';
}
echo '</td>';
</tr>

<tr>
<td></td>
<td><input type="submit" value="Finalizar" /></td>
</tr>
</table>
</form>
</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/cliente_editar.2.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$msg = 'Acesso inválido';
if(isset($_POST['nome'], $_POST['endereco'], $_POST['bairro'],
$_POST['data_nascimento'])) {
$objCliente = new Cliente();

    if(Data::validarData($_POST['data_nascimento'])) {
        $data_nascimento = Data::formataDataPTtoEN($_POST['data_nascimento']);
        $dados = array( 'nome' => $_POST['nome'],
            'endereco' => $_POST['endereco'],
            'bairro' => $_POST['bairro'],
            'data_nascimento' => $data_nascimento
        );

        $objCliente->editar($_POST['cliente_id'], $dados);
        $alterados = mysql_affected_rows();
        if($alterados > 0)
            $msg = 'Cliente alterado com sucesso';
        elseif($alterados == 0)
            $msg = 'Nada foi alterado';
        else
            $msg = 'Não foi possível efetuar a alteração do cliente. Tente
novamente.';
    }
    else {
        $msg = 'Data de nascimento inválida';
    }
}
header('Location: ' . URL_ADMIN . 'cliente_index.php?msg=' . $msg);
?>

```

php/_admin/cliente_editar.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');

$objCliente = new Cliente();
$cliente = $objCliente->get(array('cliente_id' => $_GET['cliente_id']));
if($cliente !== false){
    include('../_includes/topo.php');
    echo '

<h2>Editar cliente</h2>
<div class="espacado">
    ' . (isset($_GET['msg']) ? '<div>'.$_GET['msg'].'</div>: ' : '') . '
<form action="cliente_editar.2.php" method="post">
<input type="hidden" value="'.$_GET['cliente_id'].'" name="cliente_id" />
<table>
<tr>
<td>Nome</td>
<td><input type="text" maxlength="45" name="nome"
value="'.$cliente['nome'].'" /></td>
</tr>

<tr>
<td>CPF</td>
<td>'.$cliente['cpf'].'</td>
</tr>

<tr>
<td>Endereço</td>
<td><input type="text" maxlength="100" name="endereco"
value="'.$cliente['endereco'].'" /></td>
</tr>

<tr>
<td>Bairro</td>
<td><input type="text" maxlength="20" name="bairro"
value="'.$cliente['bairro'].'" /></td>
</tr>

<tr>
<td>Data nascimento</td>
<td><input type="text" maxlength="10" name="data_nascimento"
value="'.Data::formataDataENToPT($cliente['data_nascimento']).'" /></td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="salvar" /></td>
</tr>
</table>
</form>
</div>;
    include('../_includes/rodape.php');
}
else{
    header('Location: cliente_index.php?msg=Cliente inválido');
}
?>

```

php/_admin/cliente_index.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
echo '
<h2>Cadastrar cliente</h2>
<div class="espacado">
    '(isset($_GET['msg'])? '<div>'.$_GET['msg'].'</div>':'').'
<form action="cliente_cadastrar.php" method="post">
<table>
<tr>
<td>Nome</td>
<td><input type="text" maxlength="45" name="nome" /></td>
</tr>

<tr>
<td>CPF</td>
<td><input type="text" maxlength="11" name="cpf" /></td>
</tr>

<tr>
<td>Endereço</td>
<td><input type="text" maxlength="100" name="endereco" /></td>
</tr>

<tr>
<td>Bairro</td>
<td><input type="text" maxlength="20" name="bairro" /></td>
</tr>

<tr>
<td>Data nascimento</td>
<td><input type="text" maxlength="10" name="data_nascimento" /></td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="cadastrar" /></td>
</tr>
</table>
</form>
</div>
<h2>Lista de clientes</h2>
<div class="espacado">';
    $objCliente = new Cliente();
    $clientes = $objCliente->gets(array('ativo' => 1));
    if(mysql_num_rows($clientes) > 0 ){

        echo '<table border="1" cellpadding="5">
<tr>
<th>Nome</th>
<th>CPF</th>
<th>Endereço</th>
<th>Bairro</th>
<th>Data de Nascimento</th>
<th>Ações</th>
</tr>';

        while($cliente = mysql_fetch_assoc($clientes)){

```

```

echo '<tr>
<td>'.$cliente['nome'].'</td>
<td>'.$cliente['cpf'].'</td>
<td>'.$cliente['endereco'].'</td>
<td>'.$cliente['bairro'].'</td>
<td>'.Data::formataDataENTOPT($cliente['data_nascimento']).'</td>
<td>
<a
href="cliente_editar.php?cliente_id='.$cliente['cliente_id'].'">Editar</a><
br/>
<a href="cliente_desativar.php?cliente_id='.$cliente['cliente_id'].'"
onclick="if(confirm(\'Deseja continuar com a exclusão?\'))return true; else
return false;">Desativar cadastro</a>
</td>
</tr>';
    }

    echo '</table>';
}
echo '</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/filme_desativar.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$msg = '';
if(isset($_GET['filme_id'])){
    $objFilme = new Filme();
    if($objFilme->remover($_GET['filme_id'])){
        $msg = 'Filme removido com sucesso';
    }
    else{
        $msg = 'Não foi possível remover este filme';
    }
}
header('Location: ' . URL_ADMIN . 'filme_index.php?msg=' . $msg);
?>

```

php/_admin/relatorios_index.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');

echo '

<h2>Relatórios</h2>
<div class="espacado">
.(isset($_GET['msg'])? '<div>'.$_GET['msg'].'</div>':'').'
<ul>
<li><a href="relatorios_mais_filme.php">Filmes mais alugados</a></li>
<li><a href="relatorios_mais_genero.php">Gênero mais visto</a></li>
</ul>
</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/index.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
echo '<div class="espacado"></div>';
include('../_includes/rodape.php');
?>

```

php/_admin/locacao.5.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
if(!isset($_GET['locacao_id']) || $_GET['locacao_id'] == ''){
header('Location: locacao_index.php?msg=Locação não localizada');
}
$objLocacaoFilmes = new LocacaoFilmes();
$objFilme = new Filme();
echo '<h2>Locação - Comprovante</h2>
<div class="espacado">
<table>
<tr>
<td valign="top">Filmes locados</td>
<td>';
    $filmes = $objLocacaoFilmes->getsByLocacaoId($_GET['locacao_id']);
    $tamanho = mysql_num_rows($filmes);
if($tamanho > 0){
    echo '<table border="1" cellspacing="0"
cellpadding="5">'.PHP_EOL.'';
echo '<tr>'.PHP_EOL.'';
    $cont = 1;
    $total = 0;
    while($filme = mysql_fetch_assoc($filmes)){
        $bgcolor = ($cont % 2 == 0?'FFFFFF' : 'EFEFEF' );
        echo '<td bgcolor="#'.$bgcolor.'">'. $filme['titulo'].'</td><td
bgcolor="#'.$bgcolor.'" align="right"><b>R$ '.
Util::formatarValorEntoPT($filme['valor']).'</b></td>'.PHP_EOL.'';
        $total += $filme['valor'];
    }
}
}

```

```

        if($cont % 1 == 0 && $cont < $tamanho)
            echo '</tr>'.PHP_EOL.'<tr>';

            $cont++;
        }
        echo '</tr>';
    echo '<tr bgcolor="#FF9999"><td>Total:</td><td><b>R$ '.
    number_format($total,2,',','').</b></td></tr>';
    echo '</table>';
}
echo '</td>
</tr>

<tr>
<td></td>
<td><a href="'.URL_ADMIN.'locacao_index.php">Nova locação</a></td>
</tr>
</table>
</div>
';
include('../_includes/rodape.php');
?>

```

php/_admin/filme_editar.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$objFilme = new Filme();
$objGenero = new Genero();
$filme = $objFilme->get(array('filme_id' => $_GET['filme_id']));
if($filme !== false){
    include('../_includes/topo.php');
    echo '

<h2>Editar filme</h2>
<div class="espacado">
    '.(isset($_GET['msg'])? '<div>'.$_GET['msg'].'</div>':'').'
<form action="filme_editar.2.php" method="post">
<input type="hidden" value="'.$_GET['filme_id'].'" name="filme_id" />
<table>
<tr>
<td>Título</td>
<td><input type="text" maxlength="45" name="titulo"
value="'.$_filme['titulo'].'" /></td>
</tr>

<tr>
<td>Gênero</td>
<td>'.$objGenero->getSelect('genero', $filme['genero']).</td>
</tr>

<tr>
<td>Valor</td>
<td><input type="text" maxlength="10" name="valor"
value="'.Util::formatarValorENToPT($filme['valor']).'" /> Exemplo:
10,00</td>
</tr>

<tr>
<td></td>

```

```

<td><input type="submit" value="salvar" /></td>
</tr>
</table>
</form>

```

```

</div>';
    include('../_includes/rodape.php');
}
else{
    header('Location: filme_index.php?msg=Filme inválido');
}
?>

```

php/_admin/filme_editar.2.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$msg = 'Acesso inválido';

if(isset($_POST['titulo'], $_POST['valor'])){
    $objFilme = new Filme();

    if($_POST['genero'] != '' ){

        if(Util::validarValor($_POST['valor'])){

            $valor = Util::formatarValorPTtoEN($_POST['valor']);

            $dados = array( 'titulo' => $_POST['titulo'],
                'valor' => $valor,
                'genero' => $_POST['genero'],
                'data_publicacao' => date('Y-m-d')
            );

            $objFilme->editar($_POST['filme_id'], $dados);
            $alterados = mysql_affected_rows();

            if($alterados > 0)
                $msg = 'Filme alterado com sucesso';

            elseif($alterados == 0)
                $msg = 'Nada foi alterado';

            else
                $msg = 'Não foi possível efetuar a alteração do filme. Tente novamente.';
        }
        else{
            $msg = 'Valor inválido';
        }
    }
    else{
        $msg = 'Selecione um gênero';
    }
}
header('Location: ' . URL_ADMIN . 'filme_index.php?msg=' . $msg);
?>

```

php/_admin/filme_cadastrar.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$msg = 'Acesso inválido';
if(isset($_POST['titulo'], $_POST['valor'])){
    $objFilme = new Filme();

    if($_POST['genero'] != '' ){
        if(Util::validarValor($_POST['valor'])){

            $valor = Util::formatarValorPTtoEN($_POST['valor']);

            $dados = array( 'titulo' => $_POST['titulo'],
                'valor' => $valor,
                'genero' => $_POST['genero'],
                'data_publicacao' => date('Y-m-d')
            );

            $objFilme->inserir($dados);

            if(mysql_affected_rows() > 0)
                $msg = 'Filme adicionado com sucesso';
            else
                $msg = 'Não foi possível efetuar o cadastro do filme. Tente novamente.';
        }
        else{
            $msg = 'Valor inválido';
        }
    }
    else{
        $msg = 'Selecione um gênero';
    }
}
header('Location: ' . URL_ADMIN . 'filme_index.php?msg=' . $msg);
?>

```

php/_admin/devolucao.2.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
if(trim($_POST['cliente_id']) == ''){
header('Location: devolucao_index.php?msg=Nenhum cliente selecionado');
    exit;
}

$objFilme = new Filme();
$filmes = $objFilme->getsFilmesDevolucaoByCliente($_POST['cliente_id']);
$tamanho = mysql_num_rows($filmes);
if($tamanho < 1){
    header('Location: devolucao_index.php?msg=Nenhum filme pendente para este
cliente');
    exit;
}
include('../_includes/topo.php');
echo '<h2>Devolução - Passo 2</h2>
<div class="espacado">
<form action="devolucao.3.php" method="POST">
'.(isset($_GET['msg']) ? $_GET['msg'] : '').'
<table>
<tr>
<td>Selecione os filmes que serão devolvidos:</td>
<td>';

        echo '<table border="1" cellspacing="0" cellpadding="5">'.PHP_EOL.'';
echo '<tr>'.PHP_EOL.'';
$cont = 1;
while($filme = mysql_fetch_assoc($filmes)){
$bgcolor = ($cont % 2 == 0?'FFFFFF' : 'EFEFEF' );
    $diferenca = Data::diferenca(date("Y-m-d"),
$filme['data_devolucao']);

    if($diferenca < 0){
        $msg_diferenca = ($diferenca * -1) . ' dia(s) para a data de
devolução';
    }
    elseif($diferenca == 0){
        $msg_diferenca = 'No prazo';
    }
    else{
        $msg_diferenca = $diferenca . ' dia(s) de atraso';
    }

    echo '<td bgcolor="#'.$bgcolor.'"><label><input type="checkbox"
name="filmes[]" value="'.$filme['locacao_filmes_id'].'"
/>'.$filme['titulo'].'</label></td><td bgcolor="#'.$bgcolor.'"
align="right"><b> R$'.
Util::formatarValorENTOtoPT($filme['valor']).'</b></td><td
bgcolor="#'.$bgcolor.'">'.Data::formataDataENTOtoPT($filme['data_devolucao'])
.' ('.$msg_diferenca.')</td>'.PHP_EOL.'';
    if($cont % 1 == 0 && $cont < $tamanho)
echo '</tr>'.PHP_EOL.'<tr>';

        $cont++;
    }
echo '</tr>';

```

```

echo '</table>';

echo '</td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="Confirmar devolução" /></td>
</tr>
</table>
</form>
</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/cliente_cadastrar.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$msg = 'Acesso inválido';
if(isset($_POST['nome'], $_POST['endereco'], $_POST['bairro'],
$_POST['data_nascimento'])){
$objCliente = new Cliente();

    if(Data::validarData($_POST['data_nascimento'])){
        $data_nascimento = Data::formataDataPTtoEN($_POST['data_nascimento']);

if($_POST['cpf'] != "" && strlen($_POST['cpf']) == 11){
    $cliente = $objCliente->get(array('cpf' => $_POST['cpf'], 'ativo' =>
1));
    if($cliente === false){
        $dados = array( 'nome' => $_POST['nome'],
            'cpf' => $_POST['cpf'],
'endereco' => $_POST['endereco'],
            'bairro' => $_POST['bairro'],
            'data_nascimento' => $data_nascimento
);

        $objCliente->inserir($dados);

        if(mysql_affected_rows() > 0)
$msg = 'Cliente adicionado com sucesso';
        else
            $msg = 'Não foi possível efetuar o cadastro do cliente. Tente
novamente.';
        }
        else{
            $msg = 'CPF existente';
        }
    }
    else{
        $msg = 'CPF inválido';
    }
}
else{

```

```

        $msg = 'Data de nascimento inválida';
    }
}
header('Location: ' . URL_ADMIN . 'cliente_index.php?msg=' . $msg);
?>

```

php/_admin/relatorios_mais_filme.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');

echo '<h2>Relatório Filmes Mais alugados</h2>
<div class="espacado">';
$objFilme = new Filme();
$filmes = $objFilme->filmeMaisAlugados();
if(mysql_num_rows($filmes) > 0){
echo '<table border="1" cellpadding="5">';
    $cont = 1;
    echo '<tr>
<th>Posição</th>
<th>Qtd</th>
<th>Título</th>
<th>Renda Total</th>
</tr>';
    while($filme = mysql_fetch_assoc($filmes)){
        echo '<tr>
<td>'. $cont. '&ordm;</td>
<td>'. $filme['qtd']. '</td>
<td>'. $filme['titulo']. '</td>
<td>R$ '. Util::formatarValorENToPT($filme['renda']). '</td>
</tr>';
        $cont++;
    }

    echo '</table>';
}
echo ' </div>';
include('../_includes/rodape.php');
?>

```

php/_admin/devolucao_index.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
echo '<h2>Devolução - Passo 1</h2>';
<div class="espacado">
<form action="devolucao.2.php" method="POST">
'.(isset($_GET['msg']) ? $_GET['msg'] : '').'
<table>
<tr>
<td>Selecione o cliente</td>
<td>
<select name="cliente_id">
<option value=""></option>';

        $objCliente = new Cliente();
        $clientes = $objCliente->getsClienteDevolucao();
        while($cliente = mysql_fetch_assoc($clientes)){
            echo '<option
value="' . $cliente['cliente_id'] . '">' . $cliente['nome'] . '</option>';
        }

echo ' </select>
</td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="Prosseguir passo 2" /></td>
</tr>
</table>
</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/locacao.4.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');
if(!isset($_POST['cliente_id']) || $_POST['cliente_id'] == ''){
    header('Location: locacao_index.php?msg=Selecione um cliente');
}
elseif(!isset($_POST['filmes']) || $_POST['filmes'] == '' ||
count($_POST['filmes']) < 1){
header('Location: locacao_index.php?msg=Nenhum filme foi selecionado');
}
$objLocacao = new Locacao();
$objLocacaoFilmes = new LocacaoFilmes();
$objFilme = new Filme();
$filmes = $objFilme->getsIn(join(',', $_POST['filmes']));
$tamanho = mysql_num_rows($filmes);
$total = 0;
while($filme = mysql_fetch_assoc($filmes)){
$total += $filme['valor'];
}
$dadosLocacao = array( 'valor_total' => $total,
'cliente_id' => $_POST['cliente_id'],
'atendente_id' => $_COOKIE['atendente_id'],
'data_locacao' => date("Y/m/d H:i:s")
);
$objLocacao->inserir($dadosLocacao);
if(mysql_affected_rows() > 0){
    $locacaoId = mysql_insert_id();
    mysql_data_seek($filmes, 0);
    while($filme = mysql_fetch_assoc($filmes)){
    $dadosLocacaoFilmes = array(
        'locacao_id' => $locacaoId,
        'filme_id' => $filme['filme_id'],
        'valor' => $filme['valor'],
        'data_devolucao' => date('Y-m-
d',mktime(1,1,1,date('m'),date('d') + DIAS_LOCACAO,date('Y'))) //DIA Atual
+ DIAS_LOCACAO dias
    );
    $objLocacaoFilmes->inserir($dadosLocacaoFilmes);
    }

    header("Location: locacao.5.php?locacao_id=" . $locacaoId);
    exit;
}
header("Location: locacao_index.php?msg=Erro. Tente novamente.");
?>

```

php/_admin/filme_index.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');

$objGenero = new Genero();
echo '

<h2>Cadastrar filme</h2>
<div class="espacado">
'.(isset($_GET['msg'])? '<div>'.$_GET['msg'].'</div>':'').'
<form action="filme_cadastrar.php" method="post">
<table>
<tr>
<td>Título</td>
<td><input type="text" maxlength="45" name="titulo" /></td>
</tr>

<tr>
<td>Gênero</td>
<td>'.$objGenero->getSelect('genero').'</td>
</tr>

<tr>
<td>Valor</td>
<td><input type="text" maxlength="5" name="valor" /> Exemplo: 10,00</td>
</tr>

<tr>
<td></td>
<td><input type="submit" value="cadastrar" /></td>
</tr>
</table>
</form>
</div>
<h2>Lista de filme</h2>
<div class="espacado">';
$objFilme = new Filme();
$filmes = $objFilme->gets(array('ativo' => 1));
if(mysql_num_rows($filmes) > 0 ){

echo '<table border="1" cellpadding="5">
<tr>
<th>Título</th>
<th>Valor</th>
<th>Data de Publicação</th>
<th>Ações</th>
</tr>';

    while($filme = mysql_fetch_assoc($filmes)){
echo '<tr>
<td>'.$filme['titulo'].'</td>
<td>R$ '.$filme['valor'].'</td>
<td>'.Data::formataDataENToPT($filme['data_publicacao']).'</td>
<td>
<a href="filme_editar.php?filme_id='.$filme['filme_id'].'">Editar</a><br/>
<a href="filme_desativar.php?filme_id='.$filme['filme_id'].'"
onclick="if(confirm(\'Deseja continuar com a exclusão?\'))return true; else
return false;">Desativar cadastro</a>

```

```

</td>
</tr>';
}

    echo '</table>';
}
echo '</div>';
include('../_includes/rodape.php');
?>

```

php/_admin/relatorios_mais_genero.php

```

<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
include('../_includes/topo.php');

echo '<h2>Relatório Gênero mais alugados</h2>
<div class="espacado">';
$objFilme = new Filme();
$objGenero = new Genero();
$filmes = $objFilme->generoMaisAlugados();
if(mysql_num_rows($filmes) > 0){
    echo '<table border="1" cellpadding="5">';
    $cont = 1;
    echo '<tr>
<th>Posição</th>
<th>Qtd</th>
<th>Gênero</th>
<th>Renda Total</th>
</tr>';
    while($filme = mysql_fetch_assoc($filmes)){
        echo '<tr>
<td>'.$cont.'&ordm;</td>
<td>'.$filme['qtd'].'</td>
<td>'.$objGenero->get($filme['genero']).'</td>
<td>R$ '.Util::formatarValorENToPT($filme['renda']).'</td>
</tr>';
        $cont++;
    }

    echo '</table>';
}
echo ' </div>';
include('../_includes/rodape.php');
?>

```

php/_admin/cliente_desativar.php

```
<?php
include_once('../_config/config.php');
include_once('../_config/logado.php');
$msg = '';
if(isset($_GET['cliente_id'])){
    $objCliente = new Cliente();
    if($objCliente->remove($_GET['cliente_id'])){
        $msg = 'Cliente removido com sucesso';
    }
    else{
        $msg = 'Não foi possível remover este cliente';
    }
}
header('Location: ' . URL_ADMIN . 'cliente_index.php?msg=' . $msg);
?>
```

php/login_deslogar.php

```
<?php

//inclusao de arquivos
require_once('_config/config.php');
setcookie('atendente_id', '', COOKIE_EXPIRADO, COOKIE_PATH,COOKIE_DOMINIO);
setcookie('nome', '', COOKIE_EXPIRADO, COOKIE_PATH,COOKIE_DOMINIO);
setcookie('usuario', '', COOKIE_EXPIRADO, COOKIE_PATH,COOKIE_DOMINIO);
setcookie('senha', '', COOKIE_EXPIRADO, COOKIE_PATH,COOKIE_DOMINIO);
setcookie('checkup', '', COOKIE_EXPIRADO, COOKIE_PATH,COOKIE_DOMINIO);
header('Location: /index.php');
?>
```

7. APÊNDICE C – Código fonte Java EE

java/src/Atendente.java

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.HashMap;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServletResponse;

public class Atendente {
    public boolean login(HttpServletResponse response, String usuario, String senha){
        Database db = new Database();
        Connection conn = db.getConnection();
        try {
            PreparedStatement ps = conn.prepareStatement("SELECT * FROM atendente WHERE
usuario = ? and senha = ?");
            senha = MD5.md5(senha);
            ps.setString(1, usuario);
            ps.setString(2, senha);

            ResultSet rs = ps.executeQuery();
            System.out.println("Tamanho: " + rs.getFetchSize());
            if(rs.next()){
                Cookie cookieUsuario = new Cookie("usuario", usuario);
                Cookie cookieSenha = new Cookie("senha", senha);

                Cookie cookieNome = new Cookie("nome", rs.getString("nome"));
                Cookie cookieAtendenteId = new Cookie("atendente_id",
rs.getString("atendente_id"));

                HashMap<String, String> hm = new HashMap<String, String>();
                hm.put("usuario", usuario);
                hm.put("senha", senha);
                hm.put("nome", rs.getString("nome"));
                hm.put("atendente_id", rs.getString("atendente_id"));

                String ch = Seguranca.geraCheckup(hm);

                Cookie cookieCheckup = new Cookie("checkup", ch);

                int expira = 3600 * 5 * 1;

                cookieUsuario.setMaxAge(expira);
                response.addCookie(cookieUsuario);

                cookieSenha.setMaxAge(expira);
                response.addCookie(cookieSenha);

                cookieNome.setMaxAge(expira);
                response.addCookie(cookieNome);

                cookieAtendenteId.setMaxAge(expira);
                response.addCookie(cookieAtendenteId);

                cookieAtendenteId.setMaxAge(expira);
                response.addCookie(cookieCheckup);

                return true;
            }
            else{
                return false;
            }
        }
        catch(SQLException e){
            try {
                conn.close();
            }

```

```

        } catch (SQLException e1) {}
        e.printStackTrace();
        return false;
    }
}
}

```

java/src/Genero.java

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class Genero {
    ArrayList<String> generos = null;

    public Genero() {
        generos = new ArrayList<String>();
        generos.add("Selecione");
        generos.add(1, "Ação");
        generos.add(2, "Animação");
        generos.add(3, "Aventura");
        generos.add(4, "Chanchada");
        generos.add(5, "Catástrofe");
        generos.add(6, "Comédia");
        generos.add(7, "Comédia romântica");
        generos.add(8, "Comédia dramática");
        generos.add(9, "Comédia de ação");
        generos.add(10, "Cult");
        generos.add(11, "Documentário");
        generos.add(12, "Drama");
    }

    public ArrayList<String> getAll(){
        return generos;
    }

    public String getById(int id){
        return generos.get(id);
    }

    public ArrayList<RelatorioGenerosMaisVisto> generosMaisVistos(){

        ArrayList<RelatorioGenerosMaisVisto> filmes = new
        ArrayList<RelatorioGenerosMaisVisto>();
        Database db = new Database();
        Connection conn = db.getConnection();
        try {
            PreparedStatement ps = conn.prepareStatement("SELECT " +
                " COUNT(*) AS qtd," +
                " f.genero," +
                " SUM(lf.valor) as renda" +
                " FROM" +
                " locacao_filmes lf" +
                " INNER JOIN filme f ON f.filme_id = lf.filme_id" +
                " GROUP BY f.genero" +
                " ORDER BY COUNT(*) DESC");

            ResultSet rs = ps.executeQuery();
            int cont = 1;
            while(rs.next()){
                filmes.add(new RelatorioGenerosMaisVisto(rs.getInt("genero"),cont,
                rs.getInt("qtd"), rs.getDouble("renda")));
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}

```

```

        cont++;
    }
    conn.close();
    return filmes;
}
catch(SQLException e){
    try {
        conn.close();
    } catch (SQLException e1) {}
    e.printStackTrace();
return null;
}
}
}

```

java/src/ClienteServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.SQLException;
import java.util.ArrayList;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class ClienteServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public ClienteServlet() {
        super();
    }
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        PrintWriter out = response.getWriter() ;
        response.setContentType ("text/html");

if(!Seguranca.validaCookies(request)){
    System.out.println("Redirecionando(Usuario não autenticado)...");
String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
    response.sendRedirect( urlWithSessionID );
    return;
}

LayoutPortal objLayout = new LayoutPortal(request);
out.println(objLayout.topo());

//CONTEUDO
String parameterCliente_id = request.getParameter("cliente_id");
String parameterAcao = request.getParameter("acao");

if(parameterCliente_id == null){

    out.println(objLayout.formCliente(null, false,
request.getParameter("status")));

    Cliente c = new Cliente();
    ArrayList<Cliente> clientes = c.getAll();
    if(clientes != null){
        out.println(objLayout.listaClientes(clientes));
    }
}
else{

```

```

        try{
            int cliente_id = Integer.parseInt(parameterCliente_id);
            Cliente c = new Cliente(cliente_id);

            if(parameterAcao != null && parameterAcao.equals("remover")){
                String urlWithSessionID = "";
                try{
                    if(c.remover() > 0){
                        urlWithSessionID =
response.encodeRedirectURL("./ClienteServlet?status=Cliente removido com sucesso");
                    }
                    else{
                        urlWithSessionID =
response.encodeRedirectURL("./ClienteServlet?status=Não foi possível remover o
cliente");
                    }
                }catch(SQLException e){
                    e.printStackTrace();
                }
                urlWithSessionID = response.encodeRedirectURL("./ClienteServlet?status=Não foi
possível remover o cliente");
            }

            response.sendRedirect( urlWithSessionID );
        }
        else{

            out.println(objLayout.formCliente(c, true,""));
        }
    }catch(NumberFormatException e){
        out.println("Código inválido");
    }
}

//RODAPE
out.println(objLayout.rodape());

out.close() ;
}
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {

    response.setContentType ("text/html");

    String parameterCliente_id = request.getParameter("cliente_id");
    String parameterNome = request.getParameter("nome");
    String parameterCpf = request.getParameter("cpf");
    String parameterEndereco = request.getParameter("endereco");
    String parameterBairro = request.getParameter("bairro");
    String parameterData_nascimento = request.getParameter("data_nascimento");

    int cliente_id = -1;

    try{
        if(parameterCliente_id != null)
            cliente_id = Integer.parseInt(parameterCliente_id);
        }catch(NumberFormatException e){
            e.printStackTrace();
        }

        Cliente c = new Cliente(cliente_id, parameterNome,parameterCpf,
parameterEndereco, parameterBairro, parameterData_nascimento);
        String urlWithSessionID = "";
        if(c.salvar()){
            urlWithSessionID = response.encodeRedirectURL("./ClienteServlet?status=Cliente
salvo com sucesso");
        }
    }

```

```

        else{
            urlWithSessionID = response.encodeRedirectURL("./ClienteServlet?status=Não
foi possível efetuar o cadastro");
        }
        response.sendRedirect( urlWithSessionID );
    }
}

```

java/src/Locacao.java

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;

public class Locacao {

    ArrayList<Filme> filmes;
    int locacao_id = -1;
    double valor_total = 0;
    int cliente_id = 0;
    int atendente_id = 1;

    public Locacao(ArrayList<Filme> filmes, double valor_total, int cliente_id, int
atendente_id) {
        this.filmes = filmes;
        this.valor_total = valor_total;
        this.cliente_id = cliente_id;
        this.atendente_id = atendente_id;
    }

    public boolean salvar() throws SQLException{

        Database db = new Database();
        Connection conn = db.getConnection();
        boolean retorno = false;
        try {
            PreparedStatement ps = conn.prepareStatement("INSERT INTO
locacao(data_locacao, valor_total, cliente_id, atendente_id) VALUES(?,?,?,?)");
            String DATE_FORMAT = "yyyy-MM-dd";
            SimpleDateFormat sdf = new SimpleDateFormat(DATE_FORMAT);
            Calendar cl = Calendar.getInstance(); // today
            ps.setString(1, sdf.format(cl.getTime()));
            ps.setDouble(2, this.valor_total);
            ps.setInt(3, this.cliente_id);
            ps.setInt(4, this.atendente_id);
            int a = ps.executeUpdate();
            if(a > 0){
                ps = conn.prepareStatement("SELECT LAST_INSERT_ID() as id");
                ResultSet result = ps.executeQuery();
                if(result.next()){
                    int ultimo_id = result.getInt("id");

                    GregorianCalendar dataDevolucao = new GregorianCalendar();
                    dataDevolucao.setTime(new Date(System.currentTimeMillis()));
                    dataDevolucao.add(Calendar.DATE, 3);

                    sdf = new SimpleDateFormat(DATE_FORMAT);

```

```

Calendar c2 = dataDevolucao;

        int sucesso = 0;
        for(Filme f : this.filmes){
            ps = conn.prepareStatement("INSERT INTO locacao_filmes(data_devolucao,
valor, filme_id, locacao_id) VALUES(?,?,?,?)");
ps.setString(1, sdf.format(c2.getTime()));
            ps.setDouble(2, f.getValor());
            ps.setInt(3, f.getFilme_id());
ps.setInt(4, ultimo_id);
            sucesso += ps.executeUpdate();

        }

        if(sucesso == this.filmes.size()){
            retorno = true;
        }
    }
}
conn.close();
return retorno;

} catch (SQLException e) {
    e.printStackTrace();
    conn.close();
    return false;
}

}

}

```

java/src/RelatorioFilmeMaisAlugados.java

```

public class RelatorioFilmeMaisAlugados {

String titulo;
    int posicao;
    int qtd;
double rendaTotal;

    public RelatorioFilmeMaisAlugados(String titulo, int posicao, int qtd, double
rendaTotal) {
        this.titulo = titulo;
this.posicao = posicao;
        this.qtd = qtd;
        this.rendaTotal = rendaTotal;
    }

    public String getTitulo() {
        return titulo;
    }

    public void setTitulo(String titulo) {
        this.titulo = titulo;
    }

    public int getPosicao() {
        return posicao;
    }

    public void setPosicao(int posicao) {
        this.posicao = posicao;
    }
}

```

```

public int getQtd() {
    return qtd;
}

public void setQtd(int qtd) {
    this.qtd = qtd;
}

public double getRendaTotal() {
    return rendaTotal;
}

public void setRendaTotal(double rendaTotal) {
    this.rendaTotal = rendaTotal;
}
}

```

java/src/Seguranca.java

```

import java.util.HashMap;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServletRequest;

public class Seguranca {

    static public final String KEY = "A6MkDhvelbBD7GVFc";

    static public boolean validaCookies(HttpServletRequest request){
        Cookie[] cookies = request.getCookies();
        if(cookies != null){
            HashMap<String, String> hm = new HashMap<String, String>();

            for( int x=0; x<cookies.length; x++){
                hm.put(cookies[x].getName(), cookies[x].getValue());
            }
            if(hm.get("checkup") == null)
                return false;

            String ch = Seguranca.geraCheckup(hm);
            System.out.println("Chave nova: " + ch);
            System.out.println("chave antiga: " + hm.get("checkup"));
            if(ch.equals(hm.get("checkup")))
                return true;
        }
        return false;
    }

    static public String geraCheckup(HashMap<String, String> hm){
        if(hm.get("usuario") != null
        && hm.get("senha") != null
        && hm.get("nome") != null
        &&hm.get("atendente_id") != null ){
            return
            MD5.md5("usuario="+hm.get("usuario")+"&senha="+hm.get("senha")+"&nome="+hm.get("nome")
            e)+"&atendente_id="+hm.get("atendente_id")+");");
        }

        return null;
    }
}

```

java/src/SairServlet.java

```

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class SairServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public SairServlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        sair(request, response);
    }

    private void sair(HttpServletRequest request, HttpServletResponse response) {
        Cookie[] cookies = request.getCookies();
        if(cookies != null){
            for( int x=0; x<cookies.length; x++){
                cookies[x].setMaxAge(0);
                response.addCookie(cookies[x]);
            }
        }

        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuario
nao autenticado.");
        try {
            response.sendRedirect( urlWithSessionID );
        } catch (IOException e) {}
        return;
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        sair(request, response);
    }
}

```

java/src/MD5.java

```

import java.math.BigInteger;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

public class MD5 {
    //criptografia da senha
    static public String md5(String senha){
        String sen = "";
        MessageDigest md = null;
        try {
            md = MessageDigest.getInstance("MD5");
        } catch (NoSuchAlgorithmException e) {
            e.printStackTrace();
        }
        BigInteger hash = new BigInteger(1, md.digest(senha.getBytes()));
        sen = hash.toString(16);
        return sen;
    }
}

```

java/src/LayoutPortal.java

```

import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.http.HttpServletRequest;

public class LayoutPortal {

    HttpServletRequest request;
    public LayoutPortal(HttpServletRequest request) {
        this.request = request;
    }

    public String topo() {
        String retorno = "<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" \"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd\">" +
            "<html xmlns=\"http://www.w3.org/1999/xhtml\">" +
            "<head>" +
            "<meta http-equiv=\"Content-Type\" content=\"text/html; charset=iso-8859-1\" />" +
            "<title>XMovies</title>" +
            "<link rel=\"stylesheet\" href=\"./css/estilo.css\" />" +
            "</head>" +
            "<body>" +
            "<div id=\"topo\">" +
            "<h1>XMovies - Dumont Locadora</h1>" +
            "</div>";
        if(Seguranca.validaCookies(request)){
            retorno += "<div id=\"menu\">" +
                "<a href=\"./ClienteServlet\">Clientes</a> |" +
                "<a href=\"./FilmeServlet\">Filmes</a> |" +
                "<a href=\"./LocacaoServlet\">Locação</a> |" +
                "<a href=\"./DevolucaoServlet\">Devolução</a> |" +
                "<a href=\"./RelatoriosServlet\">Relatórios</a> |" +
                "<a href=\"./SairServlet\">Sair</a>" +
                "</div>";
        }
        retorno += "<div id=\"conteudo\">";
        return retorno;
    }

    public String rodape(){

```

```

        return "</div>" +
            "</div>" +
            "<div id=\"rodape\">" +
            "Sistema complementar da monografia \"Avaliação de produtividade PHP x Java\".
            Autor: Luis Felipe Corrêa Pérez" +
            "</div>" +
            "</body>" +
            "</html>";
    }

    public String login(){
        String retorno = "<form action=\"LoginServlet\" method=\"post\">" +
            "<table>" +
            "<tr>" +
            "<td>Usuário:</td>" +
            "<td><input type=\"text\" name=\"usuario\" maxlength=\"20\" /></td>" +
            "</tr>" +
            "<tr>" +
            "<td>Senha:</td>" +
            "<td><input type=\"password\" name=\"senha\" maxlength=\"10\"></td>" +
            "</tr>" +
            "<tr>" +
            "<td></td>" +
            "<td><input type=\"submit\" value=\"Acessar\"></td>" +
            "</tr>" +
            "</table>";
        return retorno;
    }

    public String formCliente(Cliente c, boolean edicao, String msg){
        String tituloForm = "", rotuloBotao = "";
        String nome = "", cpf = "", endereco = "", bairro = "", data_nascimento = "";

        if(edicao){
            tituloForm = "Editar";
            rotuloBotao = "Salvar";

            nome = c.getNome();
            cpf = c.getCpf();
            endereco = c.getEndereco();
            bairro = c.getBairro();
            data_nascimento = c.getData_nascimento();
        }
        else{
            tituloForm = "Cadastrar";
            rotuloBotao = "Cadastrar";
        }

        return "<h2>"+ tituloForm + " cliente</h2>" +
            "<div class=\"espacado\">" +
            "<form action=\"./ClienteServlet\" method=\"post\">" +
            (msg != null ? msg : "") +
            (edicao ? "<input type=\"hidden\" name=\"cliente_id\" "
            value=\""+c.getCliente_id()+"\" /> " : "") +
            "<table>" +
            "<tr>" +
            "<td>Nome</td>" +
            "<td><input type=\"text\" maxlength=\"45\" name=\"nome\" value=\""+nome+"\" "
            /></td>" +
            "</tr>" +
            "<tr>" +
            "<td>CPF</td>" +
            "<td><input type=\"text\" maxlength=\"11\" name=\"cpf\" value=\""+cpf+"\" "
            /></td>" +
            "</tr>" +
            "<tr>" +
            "<td>Endereço</td>" +

```

```

        "<td><input type=\"text\" maxlength=\"100\" name=\"endereco\"
value=\""+endereco+"\" /></td>" +
"</tr>" +
        "<tr>" +
        "<td>Bairro</td>" +
"<td><input type=\"text\" maxlength=\"20\" name=\"bairro\" value=\""+bairro+"\"
/></td>" +
"</tr>" +
        "<tr>" +
        "<td>Data nascimento</td>" +
        "<td><input type=\"text\" maxlength=\"10\" name=\"data_nascimento\"
value=\""+data_nascimento+"\" /></td>" +
"</tr>" +
        "<tr>" +
        "<td></td>" +
        "<td><input type=\"submit\" value=\"" + rotuloBotao + "\" /></td>" +
"</tr>" +
        "</table>" +
        "</form>" +
        "</div>";
    }

    public String listaClientes(ArrayList<Cliente> clientes){
String retorno = "<h2>Lista de clientes</h2>" +
"<div class=\"espacado\">" +
        "<table border=\"1\" cellpadding=\"5\">" +
        "<tr>" +
        "<th>Nome</th>" +
        "<th>CPF</th>" +
        "<th>Endereço</th>" +
        "<th>Bairro</th>" +
"<th>Data de Nascimento</th>" +
"<th>Ações</th>" +
        "</tr>";

        for(Cliente c : clientes){
String data = Data.formatarDataENToPT(c.getData_nascimento());
retorno += "<tr>" +
        "<td>"+c.getNome()+"</td>" +
"<td>"+c.getCpf()+"</td>" +
        "<td>"+c.getEndereco()+"</td>" +
"<td>"+c.getBairro()+"</td>" +
        "<td>"+data+"</td>" +
        "<td>" +
        "<a
href=\"ClienteServlet?cliente_id="+c.getCliente_id()+">Editar</a><br/>" +
        "<a href=\"ClienteServlet?acao=remover&cliente_id="+c.getCliente_id()+"\"
onclick=\"if(confirm(\"'Deseja continuar com a exclusão?'\'))return true; else return
false;\">Desativar cadastro</a>" +
        "</td>" +
        "</tr>";
        }

        retorno += "</table>" +
        "</div>";

return retorno;

    }

    public String formFilme(Filme f, boolean edicao, String msg){
String tituloForm = "", rotuloBotao = "";
String titulo = "";
int genero = 0;
double valor = 0;

    if(edicao){
tituloForm = "Editar";

```

```

        rotuloBotao = "Salvar";

        titulo = f.getTitulo();
        genero = f.getGenero();
        valor = f.getValor();
    }
    else{
        tituloForm = "Cadastrar";
        rotuloBotao = "Cadastrar";
    }

    String retorno = "<h2>" + tituloForm + " cliente</h2>" +
"<div class=\"espacado\">" +
        "<form action=\"./FilmeServlet\" method=\"post\">" +
        (msg != null ? msg : "") +
        (edicao ? "<input type=\"hidden\" name=\"filme_id\"
value=\""+f.getFilme_id()+"\" /> " : "") +
        "<table>" +
        "<tr>" +
            "<td>Título</td><td><input type=\"text\" maxlength=\"45\" name=\"titulo\"
value=\""+titulo+"\" /></td>" +
        "</tr>" +
        "<tr>" +
            "<td>Gênero</td>" +
        "<td>";

        Genero g = new Genero();
        ArrayList<String> generos = g.getAll();
        String select = "<select name=\"genero\" />";
        int cont = 1;
        for(String gen : generos){
            select += "<option value=\""+cont+"\" " + (cont == genero?
"selected=\"selected\"":"") + ">"+gen+"</option>";
            cont++;
        }
        select += "</select>";

        retorno += select + "</td>" +
"</tr>" +
        "<tr>" +
            "<td>Valor</td>" +
            "<td><input type=\"text\" maxlength=\"5\" name=\"valor\" value=\""+(valor == 0 ?
"": valor)+"\" /> Exemplo: 10,00</td>" +
        "</tr>" +
        "<tr>" +
            "<td></td>" +
            "<td><input type=\"submit\" value=\"" + rotuloBotao + "\" /></td>" +
        "</tr>" +
        "</table>" +
        "</form>" +
        "</div>";

    return retorno;
}

public String listaFilme(ArrayList<Filme> filmes){
    String retorno = "<h2>Lista de filme</h2>" +
"<div class=\"espacado\">" +
        "<table border=\"1\" cellpadding=\"5\">" +
        "<tr>" +
            "<th>Título</th>" +
            "<th>Valor</th>" +
            "<th>Data de Publicação</th>" +
            "<th>Ações</th>" +
        "</tr>";

        for(Filme f : filmes){
            String data = Data.formatarDataENTOtoPT(f.getData_publicacao());

```

```

        retorno += "<tr>" +
            "<td>"+f.getTitulo()+"</td>" +
            "<td>R$ "+f.getValor()+"</td>" +
            "<td>"+data+"</td>" +
            "<td>" +
            "<a href=\"FilmeServlet?filme_id=" + f.getFilme_id() +
"\">Editar</a><br/>" +
            "<a href=\"FilmeServlet?acao=remove&filme_id=" + f.getFilme_id() + "\""
onclick=\"if(confirm(\"'Deseja continuar com a exclusão?\"))return true; else return
false;\">Desativar cadastro</a>" +
            "</td>" +
            "</tr>";
    }

    retorno += "</table>" +
        "</div>";

return retorno;
}

public String form1Locacao(String mensagem){
String retorno = "<h2>Locação - Passo 1</h2>" +
    "<div class=\"espacado\">" +
    (mensagem != null ? "<p>" + mensagem + "</p>" : "")+
    "<form action=\"Locacao2Servlet\" method=\"POST\">" +
    "<table>" +
    "<tr>" +
    "<td>Selecione o cliente</td>" +
    "<td>";

    Cliente c = new Cliente();
    ArrayList<Cliente> clientes = c.getAll();
    String select = "<select name=\"cliente_id\" />";
    select += "<option value=\"\"></option>";
    for(Cliente a : clientes){
        select += "<option value=\"" + a.getCliente_id() + "\" " +
">" + a.getNome() + "</option>";
    }

    select += "</select>";

    retorno += select + "</td>" +
        "</tr>" +
        "<tr>" +
        "<td></td>" +
        "<td><input type=\"submit\" value=\"Prosseguir passo 2\" /></td>" +
        "</tr>" +
        "</table>" +
        "</form>" +
        "</div>";
return retorno;
}

public String form2Locacao(String cliente_id){
String retorno = "<h2>Locação - Passo 2</h2>" +
"<div class=\"espacado\">" +
    "<form action=\"Locacao3Servlet\" method=\"POST\">" +
    "<input type=\"hidden\" name=\"cliente_id\" value=\"" + cliente_id + "\"
/>" +
    "<table>" +
    "<tr>" +
    "<td valign=\"top\">Selecione o(s) filme(s)</td>" +
    "<td>" +
    "<table border=\"1\" cellspacing=\"0\" cellpadding=\"5\">" +
"<tr>";
    Filme f1 = new Filme();
    ArrayList<Filme> filmes = f1.getDisponiveis();
String bgcolor = "";

```

```

        int cont = 1;
        DecimalFormat a = new DecimalFormat("0.00");
        for(Filme f : filmes){
            bgcolor = bgcolor == "#EFEFEF" ? "#FFFFFF" : "#EFEFEF" ;

            retorno += "<td bgcolor=\"\" + bgcolor + "\">" +
"<label>" +
                "<input type=\"checkbox\" name=\"filmes\"
value=\""+f.getFilme_id()+"\" />" +
                f.getTitulo() +
            "</label>" +
            "</td>" +
            "<td bgcolor=\"\" + bgcolor + "\" align=\"right\">"+
a.format(f.getValor()) +"</td>";

            if(cont % 3 == 0){
                retorno += "</tr><tr>";
            }
            cont++;
        }
        retorno += "</tr>" +
"</table>" +
"</td>" +
"</tr>" +
"<tr>" +
"<td></td>" +
"<td><input type=\"submit\" value=\"Prosseguir\" /></td>" +
"</tr>" +
"</table>" +
"</form>";
        return retorno;
    }

    public String form3Locacao(String cliente_id, List<Filme> filmes){
        String retorno = "<h2>Locação - Confirmação</h2>" +
"<div class=\"espacado\">" +
            "<form action=\"Locacao4Servlet\" method=\"POST\">" +
            "<input type=\"hidden\" name=\"cliente_id\" value=\"\" + cliente_id + \"\"
/>" +
            "<table>" +
            "<tr>" +
            "<td valign=\"top\">Selecione o(s) filme(s)</td>" +
            "<td>" +
            "<table border=\"1\" cellspacing=\"0\" cellpadding=\"5\">" +
            "";

            String bgcolor = "";
            double total = 0;

            DecimalFormat a = new DecimalFormat("0.00");

            for(Filme f : filmes){
                bgcolor = bgcolor == "#EFEFEF" ? "#FFFFFF" : "#EFEFEF" ;

                retorno += "<tr>" +
                    "<td bgcolor=\"\" + bgcolor + "\">" +
                    "<label>" +
                        "<input type=\"checkbox\" name=\"filmes\" checked=\"checked\"
value=\""+f.getFilme_id()+"\" />" +
                        f.getTitulo() +
                    "</label>" +
                    "</td>" +
                    "<td bgcolor=\"\" + bgcolor + "\" align=\"right\">"+
                    a.format(f.getValor()) +"</td>" +
                    "</tr>";

                total += f.getValor();
            }
        }
    }

```

```

        retorno += " " +
"<tr bgcolor=\"#ff9999\">" +
        "<td>Total</td>" +
        "<td>"+a.format(total)+"</td>" +
        "</tr>" +
"</table>" +
"</td>" +
"</tr>" +
"<tr>" +
"<td></td>" +
"<td><input type=\"submit\" value=\"Prosseguir\" /></td>" +
"</tr>" +
"</table>" +
"</form>";

    return retorno;
}

public String form4Locacao(String cliente_id, List<Filme> filmes){
String retorno = "<h2>Locação - Comprovante</h2>" +
"<div class=\"espacado\">" +
    "<form action=\"Locacao4Servlet\" method=\"POST\">" +
    "<input type=\"hidden\" name=\"cliente_id\" value=\"" + cliente_id + "\"
/>" +
    "<table>" +
    "<tr>" +
    "<td valign=\"top\">Filmes locados</td>" +
    "<td>" +
    "<table border=\"1\" cellspacing=\"0\" cellpadding=\"5\">" +
    "";

    String bgcolor = "";
    double total = 0;

    DecimalFormat a = new DecimalFormat("0.00");

for(Filme f : filmes){
    bgcolor = bgcolor == "#EFEFEF" ? "#FFFFFF" : "#EFEFEF" ;

        retorno += "<tr>" +
            "<td bgcolor=\"" + bgcolor + "\">" +
"<label>" +

                f.getTitulo() +
                "</label>" +
                "</td>" +
                "<td bgcolor=\"" + bgcolor + "\" align=\"right\">" +
a.format(f.getValor()) + "</td>" +
"</tr>";

        total += f.getValor();
    }

    retorno += " " +
"<tr bgcolor=\"#ff9999\">" +
        "<td>Total</td>" +
        "<td>"+a.format(total)+"</td>" +
        "</tr>" +
"</table>" +
"</td>" +
"</tr>" +
"<tr>" +
"<td></td>" +
    "<td><a href=\"LocacaoServlet\">Nova locação</a></td>" +

```

```

"<td></td>" +
    "</tr>" +
    "</table>" +
    "</form>";

    return retorno;
}
public String form1Devolucao(String mensagem) {
String retorno = "<h2>Devolução - Passo 1</h2>" +
    "<div class=\"espacado\">" +
    (mensagem != null ? "<p>" + mensagem + "</p>" : "") +
    "<form action=\"Devolucao2Servlet\" method=\"POST\">" +
"<table>" +
    "<tr>" +
    "<td>Selecione o cliente</td>" +
    "<td>";

    Cliente c = new Cliente();
    ArrayList<Cliente> clientes = c.getClientesFilmesLocados();
String select = "<select name=\"cliente_id\" />";
    select += "<option value=\"\"></option>";
    for(Cliente a : clientes){
        select += "<option value=\"" + a.getCliente_id() + "\" " + ">" + a.getNome()
+ "</option>";
    }

    select += "</select>";

    retorno += select + "</td>" +
        "</tr>" +
        "<tr>" +
        "<td></td>" +
        "<td><input type=\"submit\" value=\"Prosseguir passo 2\" /></td>" +
        "</tr>" +
        "</table>" +
        "</form>" +
        "</div>";
    return retorno;
}

public String form2Devolucao(String cliente_id){
String retorno = "<h2>Devolução - Passo 2</h2>" +
"<div class=\"espacado\">" +
    "<form action=\"Devolucao3Servlet\" method=\"POST\">" +
    "<input type=\"hidden\" name=\"cliente_id\" value=\"" + cliente_id + "\"
/>" +
    "<table>" +
    "<tr>" +
    "<td valign=\"top\">Selecione o(s) filme(s)</td>" +
    "<td>" +
    "<table border=\"1\" cellspacing=\"0\" cellpadding=\"5\">" +
"<tr>";
        Filme f1 = new Filme();
        ArrayList<Filme> filmes = f1.getsFilmesDevolucaoByCliente(cliente_id);
String bgcolor = "";
        int cont = 1;
        DecimalFormat a = new DecimalFormat("0.00");
for(Filme f : filmes){
    bgcolor = bgcolor == "#EFEFEF" ? "#FFFFFF" : "#EFEFEF" ;

        retorno += "<td bgcolor=\"" + bgcolor + "\">" +
"<label>" +
            "<input type=\"checkbox\" name=\"filmes\"
value=\"" + f.getFilme_id() + "\" />" +
            f.getTitulo() +
            "</label>" +
            "</td>" +

```

```

                "<td bgcolor=\"" + bgcolor + "\"" align="right">" +
a.format(f.getValor()) + "</td>";

if(cont % 3 == 0){
    retorno += "</tr><tr>";
}
    cont++;
}
    retorno += "</tr>" +
"</table>" +
"</td>" +
"</tr>" +
"<tr>" +
"<td></td>" +
"<td><input type=\"submit\" value=\"Prosseguir\" /></td>" +
"</tr>" +
"</table>" +
"</form>";
return retorno;
}

public String form1Relatorios(){
    String retorno = "<h2>Relatórios</h2>" +
"<div class=\"espacado\">" +
"<ul>" +
"<li><a href=\"RelatoriosServlet?rel=filmes_mais_alugados\">Filmes mais
alugados</a></li>" +
"<li><a href=\"RelatoriosServlet?rel=genero_mais_visto\">Gênero mais
visto</a></li>" +
"</ul>" +
"</div>";

    return retorno;
}

public String relatorioFilmesMaisAlugados(){
    String retorno = "<h2>Relatório Filmes mais alugados</h2>" +
"<div class=\"espacado\">" +
"<table border=\"1\" cellpadding=\"5\">" +
"<tr>" +
"<th>Posição</th>" +
"<th>Qtd</th>" +
"<th>Título</th>" +
"<th>Renda Total</th>" +
"</tr>";

    Filme f = new Filme();
    ArrayList<RelatorioFilmeMaisAlugados> rel = f.maisAlugados();
    DecimalFormat format = new DecimalFormat("0.00");
    for(RelatorioFilmeMaisAlugados r : rel){
        retorno += "<tr>" +
"<td>"+r.getPosicao()+"</td>" +
"<td>"+r.getQtd()+"</td>" +
"<td>"+r.getTitulo()+"</td>" +
"<td>R$ "+format.format(r.getRendaTotal())+"</td>" +
"</tr>";
    }

    retorno += "</table>" +
"</div>";
return retorno;
}

public String relatorioGeneroMaisVisto(){
    String retorno = "<h2>Relatório Genero mais visto</h2>" +
"<div class=\"espacado\">" +
"<table border=\"1\" cellpadding=\"5\">" +
"<tr>" +

```

```

    "<th>Posição</th>" +
    "<th>Qtd</th>" +
    "<th>Genero</th>" +
    "<th>Renda Total</th>" +
    "</tr>";

    Genero g = new Genero();
    ArrayList<RelatorioGenerosMaisVisto> rel = g.generosMaisVistos();
    DecimalFormat format = new DecimalFormat("0.00");
    for(RelatorioGenerosMaisVisto r : rel){
        retorno += "<tr>" +
            "<td>"+r.getPosicao()+"</td>" +
            "<td>"+r.getQtd()+"</td>" +
            "<td>"+g.getById(r.getGenero())+"</td>" +
            "<td>R$ "+format.format(r.getRendaTotal())+"</td>" +
            "</tr>";
    }

    retorno += "</table>" +
    "</div>";
    return retorno;
}
}

```

java/src/Locacao3Servlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Locacao3Servlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public Locacao3Servlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./SairServlet");
        response.sendRedirect( urlWithSessionID );
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    if(!Seguranca.validaCookies(request)){
        System.out.println("Redirecionando(Usuario não autenticado)...");
    String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
    não autenticado");
        response.sendRedirect( urlWithSessionID );
        return;
    }
    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html" );

    System.out.println("Locacao3Servlet");

    LayoutPortal objLayout = new LayoutPortal(request);

    String parameterCliente = request.getParameter("cliente_id");
    String[] parameterFilmes = request.getParameterValues("filmes");

    if(parameterCliente == null || parameterCliente.equals("")){
        System.out.println("Redirecionando(1)...");
        String urlWithSessionID = "";
        urlWithSessionID =
    response.encodeRedirectURL("./LocacaoServlet?status=Selecione um cliente válido");
        response.sendRedirect( urlWithSessionID );
        return;
    }
    else if(parameterFilmes == null || parameterFilmes.length < 1){
        System.out.println("Redirecionando(2)...");
        String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./LocacaoServlet?status=Nenhum filme
    selecionado");
    response.sendRedirect( urlWithSessionID );
        return;
    }

    List<Filme> filmes = new ArrayList<Filme>();

    for(String s : parameterFilmes){
        Filme f = new Filme(Integer.parseInt(s));

```

```

        filmes.add(f);
    }

    out.println(objLayout.topo());
    out.println(objLayout.form3Locacao(parameterCliente, filmes));
    out.println(objLayout.rodape());
    out.close() ;
}
}

```

java/src/Locacao2Servlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Locacao2Servlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public Locacao2Servlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./SairServlet");
        response.sendRedirect( urlWithSessionID );
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
if(!Seguranca.validaCookies(request)){
        System.out.println("Redirecionando(Usuario não autenticado)...");
String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
        response.sendRedirect( urlWithSessionID );
        return;
    }
    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html") ;

    System.out.println("Locacao2Servlet");

    LayoutPortal objLayout = new LayoutPortal(request);
    String parameterCliente = request.getParameter("cliente_id");
    if(parameterCliente != null && !parameterCliente.equals("")){
        System.out.println("parameterCliente: " + parameterCliente);
        out.println(objLayout.topo());
        out.println(objLayout.form2Locacao(parameterCliente));
        out.println(objLayout.rodape());
        out.close();
    }
    else{
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID =
response.encodeRedirectURL("./LocacaoServlet?status=Selecione um cliente válido");
        response.sendRedirect( urlWithSessionID );
    }
}
}
}

```

java/src/LocacaoServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class LocacaoServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public LocacaoServlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
if(!Seguranca.validaCookies(request)){
    System.out.println("Redirecionando(Usuario não autenticado)...");
String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
    response.sendRedirect( urlWithSessionID );
    return;
}

    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html") ;
    LayoutPortal objLayout = new LayoutPortal(request);
    out.println(objLayout.topo());
    String mensagem = request.getParameter("status");
    out.println(objLayout.formLocacao(mensagem));
    out.println(objLayout.rodape());
    out.close() ;
}
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {}
}

```

java/src/RelatoriosServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
/**
 * Servlet implementation class ClienteServlet
 */
public class RelatoriosServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public RelatoriosServlet() {
        super();
    }
}
/**

```

```

    * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
    */
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
if(!Seguranca.validaCookies(request)){
    System.out.println("Redirecionando(Usuario não autenticado)...");
String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
    response.sendRedirect( urlWithSessionID );
    return;
}

    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html") ;
    LayoutPortal objLayout = new LayoutPortal(request);
    out.println(objLayout.topo());
    String rel = request.getParameter("rel");
    if(rel == null){
        out.println(objLayout.form1Relatorios());
    }
    else if(rel.equals("genero_mais_visto")){
        out.println(objLayout.relatorioGeneroMaisVisto());
    }
else if(rel.equals("filmes_mais_alugados")){
    out.println(objLayout.relatorioFilmesMaisAlugados());
}
    else{
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
urlWithSessionID = response.encodeRedirectURL("./RelatoriosServlet?status=Selecione
um relatorio válido");
response.sendRedirect( urlWithSessionID );
    }

    out.println(objLayout.rodape());
    out.close() ;
}
/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
    */
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
// TODO Auto-generated method stub
}
}

```

java/src/Cliente.java

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class Cliente {
    private int cliente_id;
    private String nome, cpf, endereco, bairro;
    private String data_nascimento;

    public Cliente() {}

    public Cliente(int cliente_id) {
        Cliente c = this.getById(cliente_id);
    }
}

```

```

        if(c != null){
            this.cliente_id = c.cliente_id;
            this.nome = c.nome;
            this.cpf = c.cpf;
        }
        this.endereco = c.endereco;
        this.bairro = c.bairro;
        this.data_nascimento = c.data_nascimento;
    }
    else{
        System.exit(-1);
    }
}

    public Cliente(int cliente_id, String nome, String cpf, String endereco, String
bairro, String data_nascimento) {
        this.cliente_id = cliente_id;
        this.nome = nome;
        this.cpf = cpf;
    }
    this.endereco = endereco;
    this.bairro = bairro;
    this.data_nascimento = data_nascimento;
}

    public Cliente(String nome, String cpf, String endereco, String bairro, String
data_nascimento) {
        this(-1, nome, cpf, endereco, bairro, data_nascimento);
    }

    public boolean existe(){
        String cpf = this.cpf;
        Database db = new Database();
        Connection conn = db.getConnection();
        try {
            PreparedStatement ps = conn.prepareStatement("SELECT cliente_id FROM cliente
WHERE cpf = ? AND ativo = 1");
            ps.setString(1, cpf);
            ResultSet rs = ps.executeQuery();

            if(rs.next()){
                conn.close();
                return true;
            }
            conn.close();
            return false;
        } catch (SQLException e) {
            try {
                conn.close();
            } catch (SQLException e1) {}
            return false;
        }
    }

    public int inserir() throws SQLException{
        if(!existe()){
            Database db = new Database();
            Connection conn = db.getConnection();
            try {
                PreparedStatement ps = conn.prepareStatement("INSERT INTO cliente(nome,
cpf, endereco, bairro, data_nascimento) VALUES(?,?,?,?,?)");
                ps.setString(1, this.nome);
                ps.setString(2, this.cpf);
                ps.setString(3, this.endereco);
                ps.setString(4, this.bairro);
                ps.setString(5, this.data_nascimento);

                int a = ps.executeUpdate();
                // System.out.println("A: " + a);
                conn.close();
                return a;
            }
        }
    }
}

```

```

    } catch (SQLException e) {
        conn.close();
        throw e;
    }
}
else{
    return -1;
}
}

public int remover() throws SQLException{
    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("UPDATE cliente " +
            "SET ativo = 0 " +
            "WHERE cliente_id = ?");
        ps.setInt(1, this.cliente_id);

        int e = ps.executeUpdate();
        conn.close();
        return e;
    } catch (SQLException e) {
        conn.close();
        throw e;
    }
}

public ArrayList<Cliente> getAll(){
    ArrayList<Cliente> clientes = new ArrayList<Cliente>();
    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("SELECT * FROM cliente WHERE
ativo = 1 ORDER BY nome ASC");
        ResultSet rs = ps.executeQuery();
        while(rs.next()){
            clientes.add(new Cliente(rs.getInt("cliente_id"), rs.getString("nome"),
rs.getString("cpf"), rs.getString("endereco"), rs.getString("bairro"),
rs.getString("data_nascimento")));
        }
        conn.close();
        return clientes;
    }
    catch(SQLException e){
        try {
            conn.close();
        } catch (SQLException e1) {}
        e.printStackTrace();
        return null;
    }
}

public ArrayList<Cliente> getClientesFilmesLocados(){
    ArrayList<Cliente> clientes = new ArrayList<Cliente>();
    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("select c.* " +
            "from cliente c " +
            "inner join locacao l on l.cliente_id = c.cliente_id " +
            "inner join locacao_filmes lf on lf.locacao_id = l.locacao_id " +
            "inner join filme f on f.filme_id = lf.filme_id " +
            "where lf.devolvido = 0 " +
            "and c.ativo = 1 " +
            "group by c.nome " +
            "order by nome asc");
    }
}

```

```

        ResultSet rs = ps.executeQuery();
        System.out.println("Analisando...");
        while(rs.next()){
            Cliente cli = new Cliente(rs.getInt("cliente_id"), rs.getString("nome"),
rs.getString("cpf"), rs.getString("endereco"), rs.getString("bairro"),
rs.getString("data_nascimento"));
            System.out.println(cli.toString());
            clientes.add(cli);
        }
        System.out.println("Fim");
        conn.close();
        return clientes;
    }
    catch(SQLException e){
        try {
            conn.close();
        } catch (SQLException e1) {}
        e.printStackTrace();
        return null;
    }
}

public Cliente getById(int cliente_id){

    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("SELECT * FROM cliente WHERE
cliente_id = ? LIMIT 1");
        ps.setInt(1, cliente_id);
        ResultSet rs = ps.executeQuery();
        while(rs.next()){
            Cliente c = new Cliente(rs.getInt("cliente_id"), rs.getString("nome"),
rs.getString("cpf"), rs.getString("endereco"), rs.getString("bairro"),
rs.getString("data_nascimento"));
            conn.close();
            return c;
        }
        conn.close();
        return null;
    }
    catch(SQLException e){
        e.printStackTrace();
        return null;
    }
}

public int atualizar() throws SQLException{
    if(existe()){
        Database db = new Database();
        Connection conn = db.getConnection();
        PreparedStatement ps = conn.prepareStatement("UPDATE cliente " +
"SET nome = ?, " +
        "cpf = ?, " +
        "endereco = ?, " +
        "bairro = ?, " +
        "data_nascimento = ?" +
        "WHERE cliente_id = ?");
        ps.setString(1, this.nome);
        ps.setString(2, this.cpf);
        ps.setString(3, this.endereco);
        ps.setString(4, this.bairro);
        ps.setString(5, this.data_nascimento);
        ps.setInt(6, this.cliente_id);
        //      System.out.println("Antes do E");
        int e = ps.executeUpdate();
        //      System.out.println("E: " + e);
    }
}

```

```

        conn.close();
        return e;
    }
    else{
        return -1;
    }
}
public boolean salvar(){
    try {
        int result = -1;
        if(this.cliente_id == -1){
            // System.out.println("Inserindo");
            result = inserir();
        }
        else{
            // System.out.println("Editando");
            result = atualizar();
        }
        if(result > 0){
            return true;
        }
        // System.out.println("Saindo...");
        return false;
    } catch (SQLException e) {
        e.printStackTrace();
        return false;
    }
}
public int getCliente_id() {
    return cliente_id;
}
public void setCliente_id(int cliente_id) {
    this.cliente_id = cliente_id;
}
public String getNome() {
    return nome;
}
public void setNome(String nome) {
    this.nome = nome;
}
public String getCpf() {
    return cpf;
}
public void setCpf(String cpf) {
    this.cpf = cpf;
}
public String getEndereco() {
    return endereco;
}
public void setEndereco(String endereco) {
    this.endereco = endereco;
}
public String getBairro() {
return bairro;
}
public void setBairro(String bairro) {
    this.bairro = bairro;
}
public String getData_nascimento() {
    return data_nascimento;
}
public void setData_nascimento(String data_nascimento) {
    this.data_nascimento = data_nascimento;
}
}

@Override

```

```

    public String toString() {
        return "-----CLIENTE-----\n" +
"Cliente ID: " + this.cliente_id + "\n" +
        "Nome      : " + this.nome + "\n" +
        "CPF       : " + this.cpf + "\n" +
        "Endereco  : " + this.endereco + "\n" +
        "Bairro    : " + this.bairro + "\n" +
        "Data Nasci: " + this.data_nascimento + "\n" +
        "-----\n";
    }
}

```

java/src/Devolucao3Servlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Devolucao3Servlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public Devolucao3Servlet() {
        super();
    }
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./SairServlet");
        response.sendRedirect( urlWithSessionID );
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
if(!Seguranca.validaCookies(request)){
        System.out.println("Redirecionando(Usuario não autenticado)...");
String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
        response.sendRedirect( urlWithSessionID );
        return;
    }

    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html") ;

    System.out.println("Devolucao3Servlet");

    String parameterCliente = request.getParameter("cliente_id");
    String[] parameterFilmes = request.getParameterValues("filmes");

    if(parameterCliente == null || parameterCliente.equals("")){
        System.out.println("Redirecionando(1)...");
        String urlWithSessionID = "";
        urlWithSessionID =
response.encodeRedirectURL("./DevolucaoServlet?status=Selecione um cliente
válido");
        response.sendRedirect( urlWithSessionID );
        return;
    }
    else if(parameterFilmes == null || parameterFilmes.length < 1){
        System.out.println("Redirecionando(2)...");
        String urlWithSessionID = "";

```

```
urlWithSessionID = response.encodeRedirectURL("./DevolucaoServlet?status=Nenhum
filme selecionado");
response.sendRedirect( urlWithSessionID );
    return;
}

int sucesso=0, fail=0;

for(String s : parameterFilmes){
    Filme f = new Filme(Integer.parseInt(s));

    if(f.devolver()){
        sucesso++;
    }
    else{
        fail++;
    }
}

    System.out.println("Redirecionando(4)...");
String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./DevolucaoServlet?status=" +
sucesso + " filmes devolvidos e " + fail + " não devolvidos");

response.sendRedirect( urlWithSessionID );
    out.close();
    return;
}
}
```

java/src/Filme.java

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class Filme {
    private int filme_id;
    private String titulo, data_publicacao;
    private double valor;
    private int genero;

    public Filme() {}

    public Filme(int filme_id) {
        Filme f = this.getById(filme_id);
        if(f != null){
            this.filme_id = f.filme_id;
this.titulo = f.titulo;
            this.data_publicacao = f.data_publicacao;
            this.valor = f.valor;
            this.genero = f.genero;
        }
        else{
            System.out.println("Finalizado porque não existe filme com este id");
System.exit(-1);
        }
    }

    public Filme(int filme_id, String titulo, String data_publicacao, double valor,
int genero) {
this.filme_id = filme_id;
        this.titulo = titulo;
        this.data_publicacao = data_publicacao;
        this.valor = valor;
        this.genero = genero;
    }

    public Filme(String titulo, String data_publicacao, double valor, int genero) {
this(-1, titulo, data_publicacao, valor, genero);
    }

    public int remover() throws SQLException{
        Database db = new Database();
        Connection conn = db.getConnection();
        try {
            PreparedStatement ps = conn.prepareStatement("UPDATE filme " +
                "SET ativo = 0 " +
                "WHERE filme_id = ?");
            ps.setInt(1, this.filme_id);
            return ps.executeUpdate();
        } catch (SQLException e) {
            throw e;
        }
    }

    public ArrayList<Filme> getAll(){
        ArrayList<Filme> filmes = new ArrayList<Filme>();
        Database db = new Database();
        Connection conn = db.getConnection();
        try {
            PreparedStatement ps = conn.prepareStatement("SELECT * FROM filme WHERE ativo
= 1 ORDER BY titulo ASC");
            ResultSet rs = ps.executeQuery();
            while(rs.next()){

```

```

        filmes.add(new Filme(rs.getInt("filme_id"),
rs.getString("titulo"),rs.getString("data_publicacao"),rs.getDouble("valor"),rs.get
Int("genero")));
    }
    conn.close();
    return filmes;
}
catch(SQLException e){
    try {
        conn.close();
    } catch (SQLException e1) {}
    e.printStackTrace();
    return null;
}
}

public ArrayList<Filme> getsFilmesDevolucaoByCliente(String cliente_id){
    ArrayList<Filme> filmes = new ArrayList<Filme>();
    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("select
f.*,locacao_filmes_id,data_devolucao, c.cliente_id" +
        " from filme f" +
        " inner join locacao_filmes lf on lf.filme_id = f.filme_id" +
        " inner join locacao l on l.locacao_id = lf.locacao_id" +
        " inner join cliente c on c.cliente_id = l.cliente_id" +
        " where c.cliente_id = ?" +
        " and lf.devolvido = 0");

        ps.setString(1, cliente_id);

        ResultSet rs = ps.executeQuery();
        while(rs.next()){
            filmes.add(new Filme(rs.getInt("filme_id"),
rs.getString("titulo"),rs.getString("data_publicacao"),rs.getDouble("valor"),rs.get
Int("genero")));
        }
        conn.close();
        return filmes;
    }
    catch(SQLException e){
        try {
            conn.close();
        } catch (SQLException e1) {}
        e.printStackTrace();
        return null;
    }
}

public ArrayList<Filme> getDisponiveis(){
    ArrayList<Filme> filmes = new ArrayList<Filme>();
    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("select f.* " +
        "from filme f " +
        "where filme_id not in (select filme_id " +
        "from locacao_filmes " +
        "where devolvido=0) " +
        "order by titulo asc");
        ResultSet rs = ps.executeQuery();
        while(rs.next()){
            filmes.add(new Filme(rs.getInt("filme_id"),
rs.getString("titulo"),rs.getString("data_publicacao"),rs.getDouble("valor"),rs.get
Int("genero")));
        }
    }
}

```

```

    }
    conn.close();
    return filmes;
}
catch(SQLException e){
    try {
        conn.close();
    } catch (SQLException e1) {}
    e.printStackTrace();
    return null;
}
}

public Filme getById(int filme_id){

    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("SELECT * FROM filme WHERE
filme_id = ? LIMIT 1");
        ps.setInt(1, filme_id);
        ResultSet rs = ps.executeQuery();
        while(rs.next()){
            Filme c = new Filme(rs.getInt("filme_id"), rs.getString("titulo"),
rs.getString("data_publicacao"), rs.getDouble("valor"), rs.getInt("genero"));
            conn.close();
            return c;
        }
        conn.close();
        return null;
    }
    catch(SQLException e){
        e.printStackTrace();
        return null;
    }
}

public int inserir() throws SQLException{
    Database db = new Database();
    Connection conn = db.getConnection();

    PreparedStatement ps = conn.prepareStatement("INSERT INTO filme(titulo, genero,
valor, data_publicacao) VALUES(?,?,?,?)");
    ps.setString(1, this.titulo);
    ps.setInt(2, this.genero);
    ps.setDouble(3, this.valor);
    ps.setString(4, this.data_publicacao);
    int a = ps.executeUpdate();
    System.out.println("A: " + a);
    conn.close();
    return a;
}

public int atualizar() throws SQLException{
    Database db = new Database();
    Connection conn = db.getConnection();

    PreparedStatement ps = conn.prepareStatement("UPDATE filme " +
"SET titulo = ?, " +
"genero = ?, " +
"valor = ?, " +
"data_publicacao = ?" +
"WHERE filme_id = ?");
    ps.setString(1, this.titulo);
    ps.setInt(2, this.genero);
    ps.setDouble(3, this.valor);
}

```

```

ps.setString(4, this.data_publicacao);

ps.setInt(5, this.filme_id);

System.out.println("LOG: " + ps.toString());

int i = ps.executeUpdate();
System.out.println("I: " + i);
conn.close();
return i;
}
public boolean salvar(){
    try {
        int result = -1;
        if(this.filme_id == -1){
            System.out.println("Inserindo");
            result = inserir();
        }
        else{
System.out.println("Atualizando");
            result = atualizar();
        }
        if(result > 0){
            return true;
        }
        return false;
    } catch (SQLException e) {
        e.printStackTrace();
        return false;
    }
}
public int getFilme_id() {
    return filme_id;
}
public void setFilme_id(int filme_id) {
    this.filme_id = filme_id;
}
public String getTitulo() {
    return titulo;
}
public void setTitulo(String titulo) {
    this.titulo = titulo;
}
public String getData_publicacao() {
    return data_publicacao;
}
public void setData_publicacao(String data_publicacao) {
    this.data_publicacao = data_publicacao;
}
public double getValor() {
    return valor;
}
public void setValor(double valor) {
    this.valor = valor;
}
public int getGenero() {
    return genero;
}
public void setGenero(int genero) {
    this.genero = genero;
}

@Override
public String toString() {
return "Titulo: " + this.titulo + "\n" +
        "Genero: " + this.genero + "\n" +
        "Valor: " + this.valor;
}

```

```

}
public boolean devolver() {
    Database db = new Database();
    Connection conn = db.getConnection();

    try {
        PreparedStatement ps = conn.prepareStatement("UPDATE locacao_filmes SET
devolvido = 1 WHERE filme_id = ?");
        ps.setInt(1, this.filme_id);
        int i = ps.executeUpdate();
        System.out.println("I: " + i);
        conn.close();
        if(i>0)
            return true;

        return false;
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return false;
}

public ArrayList<RelatorioFilmeMaisAlugados> maisAlugados(){
    ArrayList<RelatorioFilmeMaisAlugados> filmes = new
ArrayList<RelatorioFilmeMaisAlugados>();
    Database db = new Database();
    Connection conn = db.getConnection();
    try {
        PreparedStatement ps = conn.prepareStatement("SELECT " +
            " COUNT(*) AS qtd," +
            " lf.filme_id," +
" f.titulo," +
            " SUM(lf.valor) as renda" +
" FROM" +
            " locacao_filmes lf" +
            " INNER JOIN filme f ON f.filme_id = lf.filme_id" +
            " GROUP BY lf.filme_id" +
            " ORDER BY COUNT(*) DESC");
        ResultSet rs = ps.executeQuery();
        int cont = 1;
        while(rs.next()){
            filmes.add(new RelatorioFilmeMaisAlugados(rs.getString("f.titulo"),cont,
rs.getInt("qtd"), rs.getDouble("renda")));
            cont++;
        }
        conn.close();
        return filmes;
    }
    catch(SQLException e){
        try {
            conn.close();
        } catch (SQLException e1) {}
        e.printStackTrace();
    }
    return null;
}
}
}

```

java/src/FilmeServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.SQLException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class FilmeServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        if(!Seguranca.validaCookies(request)){
            System.out.println("Redirecionando(Usuario não autenticado)...");
            String urlWithSessionID = "";
            urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
            não autenticado");
            response.sendRedirect( urlWithSessionID );
            return;
        }

        PrintWriter out = response.getWriter() ;
        response.setContentType ("text/html");
        LayoutPortal objLayout = new LayoutPortal(request);
        out.println(objLayout.topo() );

        //CONTEUDO
        String parameterFilme_id = request.getParameter("filme_id");
        String parameterAcao = request.getParameter("acao");

        if(parameterFilme_id == null){

            out.println(objLayout.formFilme(null, false,
            request.getParameter("status")));

        Filme f = new Filme();
        ArrayList<Filme> filmes = f.getAll();
        if(filmes != null){
            out.println(objLayout.listaFilme(filmes));
        }
        else{
            try{
                int filme_id = Integer.parseInt(parameterFilme_id);
                Filme f = new Filme(filme_id);

                if(parameterAcao != null && parameterAcao.equals("remove")){
                    String urlWithSessionID = "";
                    try{
                        if(f.remove() > 0){
                            urlWithSessionID = response.encodeRedirectURL("./FilmeServlet?status=Filme removido
                            com sucesso");
                        }
                    }
                    else{
                        urlWithSessionID =
                            response.encodeRedirectURL("./FilmeServlet?status=Não foi possível remover o
                            filme");
                    }
                }
            }catch(SQLException e){
                e.printStackTrace();
            }
        }
    }
}

```

```

urlWithSessionID = response.encodeRedirectURL("./FilmeServlet?status=Não foi
possível remover o filme");
    }
    response.sendRedirect( urlWithSessionID );
}
else{
    out.println(objLayout.formFilme(f, true,""));
}
}catch(NumberFormatException e){
    out.println("Código inválido");
}
}

//RODAPE
out.println(objLayout.rodape());

    out.close() ;
}
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
if(!Seguranca.validaCookies(request)){
    System.out.println("Redirecionando(Usuario não autenticado)...");
    String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
    response.sendRedirect( urlWithSessionID );
    return;
}

response.setContentType ("text/html");

String parameterFilme_id = request.getParameter("filme_id");
String parameterTitulo = request.getParameter("titulo");
String parameterGenero = request.getParameter("genero");
String parameterValor = request.getParameter("valor");

int filme_id = -1, genero = -1;
double valor = 0;

System.out.println("To aqui");

    try{
        if(parameterFilme_id != null)
filme_id = Integer.parseInt(parameterFilme_id);

genero = Integer.parseInt(parameterGenero);
        valor = Double.parseDouble(parameterValor);
    }catch(NumberFormatException e){
        e.printStackTrace();
    }

    Date data = new Date();
    SimpleDateFormat formatador = new SimpleDateFormat("yyyy-MM-dd");
String data2 = formatador.format(data);

    Filme f = new Filme(filme_id,parameterTitulo, data2 , valor, genero);
    f.setGenero(genero);
    f.setTitulo(parameterTitulo);
f.setValor(valor);

    String urlWithSessionID = "";

    if(f.salvar()){
urlWithSessionID = response.encodeRedirectURL("./FilmeServlet?status=Filme salvo
com sucesso");

```

```
    }
    else{
        urlWithSessionID = response.encodeRedirectURL("./FilmeServlet?status=Não foi
possível efetuar o cadastro");
    }

    response.sendRedirect( urlWithSessionID );
}
}
```

java/src/Locacao4Servlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.SQLException;
import java.util.ArrayList;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Locacao4Servlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public Locacao4Servlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./SairServlet");
        response.sendRedirect( urlWithSessionID );
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    if(!Seguranca.validaCookies(request)){
        System.out.println("Redirecionando(Usuario não autenticado)...");
    String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
    não autenticado");
        response.sendRedirect( urlWithSessionID );
        return;
    }

    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html" ) ;

    System.out.println("Locacao4Servlet");

    LayoutPortal objLayout = new LayoutPortal(request);

    String parameterCliente = request.getParameter("cliente_id");
    String[] parameterFilmes = request.getParameterValues("filmes");

    if(parameterCliente == null || parameterCliente.equals("")){
        System.out.println("Redirecionando(1)...");
        String urlWithSessionID = "";
        urlWithSessionID =
    response.encodeRedirectURL("./LocacaoServlet?status=Selecione um cliente válido");
        response.sendRedirect( urlWithSessionID );
        System.exit(-1);
    }
    else if(parameterFilmes == null || parameterFilmes.length < 1){
        System.out.println("Redirecionando(2)...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LocacaoServlet?status=Nenhum filme
    selecionado");
        response.sendRedirect( urlWithSessionID );
        return;
    }

    ArrayList<Filme> filmes = new ArrayList<Filme>();

    double total = 0;

```

```

for(String s : parameterFilmes){
    Filme f = new Filme(Integer.parseInt(s));
    total+=f.getValor();
    filmes.add(f);
    System.out.println("Total: " + total);
}

Locacao l = new Locacao(filmes, total, Integer.parseInt(parameterCliente), 1);

try {
    if(l.salvar()){
        out.println(objLayout.topo());
        out.println(objLayout.form4Locacao(parameterCliente, filmes));
        out.println(objLayout.rodape());
        out.close() ;
        return;
    }
} catch (SQLException e) {
    System.out.println("Redirecionando(3)...");
    String urlWithSessionID = "";
    urlWithSessionID = response.encodeRedirectURL("./LocacaoServlet?status=Falha
na locacao");
    out.close();
    response.sendRedirect( urlWithSessionID );
    return;
}
}
}

```

java/src/DevolucaoServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class DevolucaoServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public DevolucaoServlet() {
        super();
        // TODO Auto-generated constructor stub
    }
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        if(!Seguranca.validaCookies(request)){
            System.out.println("Redirecionando(Usuario não autenticado)...");
            String urlWithSessionID = "";
            urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
            não autenticado");
            response.sendRedirect( urlWithSessionID );
            return;
        }

        PrintWriter out = response.getWriter() ;
        response.setContentType ("text/html") ;
        LayoutPortal objLayout = new LayoutPortal(request);
        out.println(objLayout.topo());
        String mensagem = request.getParameter("status");
        out.println(objLayout.form1Devolucao(mensagem));
        out.println(objLayout.rodape());
        out.close() ;
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        // TODO Auto-generated method stub
    }
}

```

java/src/LoginServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
/**
 * Servlet implementation class LoginServlet
 */
public class LoginServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    /**
     * @see HttpServlet#HttpServlet()
     */
    public LoginServlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        response.setContentType ("text/html");
        LayoutPortal objLayout = new LayoutPortal(request);
        out.println(objLayout.topo());
        out.println(objLayout.login());
        out.println(objLayout.rodape());
        out.close() ;
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        response.setContentType ("text/html");
        String parameterUsuario = request.getParameter("usuario");
        String parameterSenha = request.getParameter("senha");
        if(parameterUsuario != null && parameterSenha != null){
            Atendente a = new Atendente();
            if(a.login(response, parameterUsuario, parameterSenha)){
                response.sendRedirect("ClienteServlet");
                return;
            }
        }
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuario ou
senha invalidos.");
        response.sendRedirect( urlWithSessionID );
        return;
    }
}

```

java/src/Data.java

```
public class Data {  
  
    public static String formatarDataENTOPT(String dt) {  
        String[] dte = dt.split("-");  
        return dt = dte[2] + "/" + dte[1] + "/" + dte[0];  
    }  
}
```

java/src/Database.java

```
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.SQLException;  
  
public class Database {  
    static final String DATABASE_URL = "jdbc:mysql://localhost:3306/locafilmes";  
    static final String USUARIO = "root";  
    static final String SENHA = "123456";  
  
    private Connection connection = null;  
  
    public Database() {  
        try {  
            connection = DriverManager.getConnection(DATABASE_URL, USUARIO, SENHA);  
        } catch (SQLException e) {  
            e.printStackTrace();  
            System.out.println("Finalizando...");  
            System.exit(-1);  
        }  
    }  
  
    public Connection getConnection() {  
        return connection;  
    }  
}
```

java/src/Devolucao2Servlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Devolucao2Servlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public Devolucao2Servlet() {
        super();
    }
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./SairServlet");
        response.sendRedirect( urlWithSessionID );
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {

if(!Seguranca.validaCookies(request)){
        System.out.println("Redirecionando(Usuario não autenticado)...");
String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./LoginServlet?status=Usuário
não autenticado");
        response.sendRedirect( urlWithSessionID );
        return;
    }

    PrintWriter out = response.getWriter() ;
    response.setContentType ("text/html") ;

    System.out.println("DevolucaoServlet");

    LayoutPortal objLayout = new LayoutPortal(request);
    String parameterCliente = request.getParameter("cliente_id");
    if(parameterCliente != null && !parameterCliente.equals("")){
        System.out.println("parameterCliente: " + parameterCliente);
        out.println(objLayout.topo());
        out.println(objLayout.form2Devolucao(parameterCliente));
        out.println(objLayout.rodape());
        out.close();
    }
    else{
        System.out.println("Redirecionando...");
        String urlWithSessionID = "";
        urlWithSessionID = response.encodeRedirectURL("./DevolucaoServlet?status=Selecione
um cliente válido");
        response.sendRedirect( urlWithSessionID );
    }
}
}
}

```

java/src/RelatorioGenerosMaisVisto.java

```
public class RelatorioGenerosMaisVisto {
    int genero;
    int posicao;
    int qtd;
    double rendaTotal;

    public RelatorioGenerosMaisVisto(int genero, int posicao, int qtd, double
rendaTotal) {
this.genero = genero;
    this.posicao = posicao;
    this.qtd = qtd;
    this.rendaTotal = rendaTotal;
    }
    public int getGenero() {
    return genero;
    }
    public void setGenero(int genero) {
    this.genero = genero;
    }
    public int getPosicao() {
    return posicao;
    }
    public void setPosicao(int posicao) {
    this.posicao = posicao;
    }
    public int getQtd() {
    return qtd;
    }
    public void setQtd(int qtd) {
    this.qtd = qtd;
    }
    public double getRendaTotal() {
    return rendaTotal;
    }
    public void setRendaTotal(double rendaTotal) {
this.rendaTotal = rendaTotal;
    }
}
```